A TEMPORAL FRAMEWORK FOR HYPERGAME ANALYSIS OF CYBER PHYSICAL SYSTEMS IN CONTESTED ENVIRONMENTS

DISSERTATION

Nicholas S. Kovach

AFIT-ENG-DS-16-J-006

AFIT-ENG-DS-16-J-006

A TEMPORAL FRAMEWORK FOR HYPERGAME ANALYSIS OF CYBER PHYSICAL SYSTEMS IN CONTESTED ENVIRONMENTS

DISSERTATION

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

Nicholas S. Kovach, B.S., M.S.

June 2016

AFIT-ENG-DS-16-J-006

A TEMPORAL FRAMEWORK FOR HYPERGAME ANALYSIS OF

CYBER PHYSICAL SYSTEMS IN CONTESTED ENVIRONMENTS

Nicholas S. Kovach, B.S., M.S.

Committee Membership:

Gary B. Lamont, PhD
Chairman

Mark E. Oxley, PhD
Member

Gilbert L. Peterson, PhD
Member

ADEDJI B. BADIRU, PhD
Dean, Graduate School of Engineering and Management

AFIT-ENG-DS-16-J-006

Abstract

Game theory is used to model conflicts between one or more players over resources. It offers players a way to reason, allowing rationale for selecting strategies that avoid the worst outcome. Game theory lacks the ability to incorporate advantages one player may have over another player. A meta-game, known as a hypergame, occurs when one player does not know or fully understand all the strategies of a game. Hypergame theory builds upon the utility of game theory by allowing a player to outmaneuver an opponent, thus obtaining a more preferred outcome with higher utility. Recent work in hypergame theory has focused on normal form static games that lack the ability to encode several realistic strategies. One example of this is when a player's available actions in the future is dependent on his selection in the past. This work presents a temporal framework for hypergame models. This framework is the first application of temporal logic to hypergames and provides a more flexible modeling for domain experts. With this new framework for hypergames, the concepts of trust, distrust, mistrust, and deception are formalized. While past literature references deception in hypergame research, this work is the first to formalize the definition for hypergames. As a demonstration of the new temporal framework for hypergames, it is applied to classical game theoretical examples, as well as a complex supervisory control and data acquisition (SCADA) network temporal hypergame. The SCADA network is an example includes actions that have a temporal dependency, where a choice in the first round affects what decisions can be made in the later round of the game. The demonstration results show that the framework is a realistic and flexible modeling method for a variety of applications.

iv

To my family and friends, sine quibus non...

Acknowledgments

I would like to express my deep gratitude to my advisor and friend, Professor Gary Lamont, for being supportive throughout all these years. A special thank you to Dr. Peterson and Dr. Oxley for reviewing my research and providing suggestions and guidance. I would be remiss without thanking my wife, for her support, encouragement, and all of the "when will you finish?" questions. Finally, to my parents who started me on this journey so many years ago.

Nicholas S. Kovach

Table of Contents

List of Figures

xiv

List of Tables

List of Acronyms

Acronym	Definition
BG	Base Game
CAP	Conflict Analysis Program
CMS	Column Mixed Strategy
CPS	Cyber Physical Systems
DoD	Department of Defense
EU	Expected Utility
FHQ	Sequential Stable
GIG	Global Information Grid
GMR	General Metarational
H	Hypergame
HAT	HNF Analysis Tool
HCCAS	Hypergame Cooperative Conflict Analysis System
HEU	Hypergame Expected Utility
HML	Hypergame Markup Language
HN	Hyper Nash equilibrium
HNF	Hypergame Normal Form
HPM	Hypergame Perception Model
ICS	Industrial Control System
ICT	Information-Communication Technology
IoT	Internet of Things
IP	Internet Protocol
ISE	Infect Support Equipment
ISP	Internet Service Provider

Acronym	Definition
MDPs	Markov decision processes
MO	Modeling Opponent
MOEA	Multi-objective evolutionary algorithm
MPE	Markov perfect equilibrium
NEMS	Nash Equilibrium Mixed Strategy
PCs	Personal Computers
PS	Pick Subgame
QPTL	Quantified Propositional Temporal Logic
RMS	Row Mixed Strategy
SCADA	Supervisory Control and Data Acquisition
SHN	Stable hyper Nash equilibrium
SMR	Symmetric Metarational
SPA	Security Policy Assistant
SPE	Subgame-perfect equilibrium
SPNE	subgame perfect Nash equilibrium
TCP	Transmission Control Protocol
TL	Temporal Logic
U.S.	United States
UI	Unilateral Improvement
UIs	Unilateral Improvements
USA	United States of America
USSR	Union of Soviet Socialist Republics
WS	Weighted Subgame
WWII	World War II
XML	Extensible Markup Language

A TEMPORAL FRAMEWORK FOR HYPERGAME ANALYSIS OF CYBER PHYSICAL SYSTEMS IN CONTESTED ENVIRONMENTS

I. Introduction

> If you know the enemy and know yourself,
>
> you need not fear the result of a hundred battles.
>
> If you know yourself but not the enemy,
>
> for every victory gained you will also suffer a defeat.
>
> If you know neither the enemy nor yourself,
>
> you will succumb in every battle.

> \- Sun Tzu [350]

Americans have been using a form of "cyber" since the 1840's, with the invention of the telegraph. While the telegraph did not present the same level of threat, as our information systems do today, it still had privacy, authentication, and physical security concerns. These same concerns still apply to technology in use today, except our dependence on the devices has increased, to the point where almost every aspect of everyday life is touched by cyber.

Life's dependence on technology and interconnected devices requires advanced diligence in order to protect the devices and technology, and life in general, from the advanced cyber threats of today. Diligence must not only be given to the capabilities and motivation of our adversaries, but also to our own capabilities, motivation, and

vulnerabilities. This need has been demonstrated through the many comtemporary cyber threats against the government (and its contractors) [1, 8, 142, 197, 237, 249], the banking industry [3, 219, 226, 309], and other vital resources [146, 236, 273]. Often these processes are controlled by interconnected devices, also known as Cyber Physical Systems (CPS).

Cyber Physical Systems (CPS) pose a bigger risk than the telegraph did over a century ago [295]. These devices are responsible for braking cars and trucks, as well as the luxury controls such as radio, air conditioning, and heat. Airplanes rely on these devices for navigation, maintenance, and control of flight surfaces. Common household appliances are interconnected, allowing for lower energy consumption or ensuring your favorite coffee is automatically prepared in the morning. Power and energy services also use Cyber Physical Systems (CPS) to control all aspects of function forming a smart grid.

1.1 Motivation

Knowing yourself and the enemy is becoming more important in securing and defending critical electronic assets, such as CPS. As each CPS is connected to the Global Information Grid (GIG), it is necessary to protect the device from cyber threats. According to Clark *et al.* [74], cyber "defenders are losing the cyber security arms race" and defenders have been approaching the problem of cyber security the wrong way. Defense strategies are often outdated, based on assumptions that no longer reflect real world threats, the attacker's capabilities, or vulnerabilities (current attack surface).

1.1.1 *Patch and Pray vs. Offensive Approach.*

Often defenders use the "patch and pray" approach to system security. This approach involves waiting for a system to be attacked, analyzing the attack, and then implementing strategies or policies to eliminate or mitigate future similar attacks.

Antivirus software based on signatures or fingerprinting, is part of the "patch and pray" approach, offering decent protection against known attack from the past, but offering little to no protection against zero-day attacks (never seen before) or future attacks. To better protect systems and improve security, it is necessary to take an offensive approach to defense allowing the defender to better understand themselves and the enemies they face, as well as the interactions between each participant.

Validating the need to find an offensive approach to defense is reflected in government-sponsored research and directives. In 2006, the federal Plan for Cyber Security and Information Assurance Research and Development [207] highlighted the trend of espionage from industrial and state-sponsored groups. In 2008, the Center for Strategic and International Studies produced the report, Securing Cyberspace for the 44th Presidency [285] which emphasized espionage and included new threats to digital intellectual property. The United States (U.S.) conducted a cyberspace policy review in 2009 which concluded the nation is falling behind in terms of cybersecurity and failed to keep pace with the growing threat [302]. In 2013, President Barack Obama stated in an executive order, "[C]yber threat to critical infrastructure continues to grow and represents one of the most serious national security challenges we must confront." [262] This executive order placed an emphasis on identifying and improving the cyber security of critical infrastructure. President Obama also signed a policy directive in 2013, stating the government should take proactive steps to "reduce vulnerabilities, minimize consequences, identify and disrupt threats, and hasten response and recovery efforts related to critical infrastructure" [261].

In 2015, President Obama recognized the importance of sharing data of cyber security risks and incidents between private companies, nonprofit organization, and federal government agencies in an executive order [264]. The President went even further in another executive order by declaring a national emergency to deal with

malicious cyber-enabled activities originating from persons outside the U.S. [263]. Both of these concepts are reflected in the Department of Defense (DoD) Cyber Strategy published in April 2015. The cyber strategy focuses on three main concepts [84]:

- Build and maintain forces and capabilities to conduct cyberspace operations.

- Defend DoD information networks and secure data.

- Prepare to defend U.S. from disruptive or destructive cyber attacks.

These reports highlight the need for improved cyber security models for a better understanding of how to protect mission critical assets. While these reports show an interest in cyber and homeland security by the U.S. since 2006, continued improvement is required as adversaries continue to develop new methods and techniques for attacking critical assets.

1.1.2   Cyber Physical System Attacks.

Around July 2010, Stuxnet was found in Supervisory Control and Data Acquisition (SCADA) systems; but not just any SCADA system - Iran's Nuclear SCADA systems [323]. Stuxnet did not steal, manipulate, or ease information, as in standard espionage [215]. Instead Stuxnet was designed to physically destroy a military target. The goal was to physically destroy a nuclear power plant fuel refinement systems through the SCADA control systems. Until this point, it was largely believed that SCADA systems were immune to attack through isolation [97]. Stuxnet has been called the first cyberwarfare weapon [98, 215]. It was discovered in Belarus by security firms [73].

Cyber-security was again put in the spot-light in September 2011 when Duqu was discovered. The purpose of Duqu is to collect data and digital assets for intelligence from industrial infrastructure and system manufacturers [345]. This information can

be used to mount a future attack using the collected information. It contained a remote access Trojan and a key logger, but could not self-replicate. It was highly targeted against specific organizations, using phishing emails with malicious Microsoft Office documents attached. Threats like Duqu provide the foundation for attacks like Stuxnet, quietly collecting sensitive information, that can be used to customized an attack with larger amounts of damage.

However, Duqu and Stuxnet appear to be just the beginning of a round of cyber-warfare weapons. Flame quickly followed Duqu and Stuxnet. Flame also known as sKyWIper, may have been in the wild for 5 to 8 years before discovery in May 2012 [80]. Flame steals information and is self propagating using multiple methods which are configurable by the attacker. It uses the keyboard, screen, microphone, storage devices, network, WiFi, Bluetooth, and USB to gather data on digital assets. These digital assets could then be infiltrated out of an organization and used to inflict damage.

Red October was discovered in October of 2012, and is believed to have been in the wild for over 5 years [208]. The malware targeted government and scientific research organizations in order to gather data and digital assets. Some assets, such as credentials were later reused, when the attacker needed to guess passwords on the network [210]. This attack was not just limited to Personal Computers (PCs) but also inflected mobile devices. This increased the amount of desired data available to attackers by opening the door to attacking to new forms of data generation, such as phone calls, text message, and other personal data.

APT28, belonging to the family called CHOPSTICK, targets critical information related to governments, military, and security organizations. Samples of APT28 were discovered from mid-2007 to September 2014. It is believed this information is likely to benefit the Russian government. APT28 appears to have a professional development

team, with standard working hours (8AM to 6PM) between Monday and Friday [99]. This indicates substantial financial backing over seven years and long term dedication to espionage against military and government targets.

In 2015, it was discovered Adobe Flash being used to infect computers then hold the computers for ransom or redirect internet traffic [333]. Attacker's purchased advertising space on websites and after a user visited a site with the ads, malware is downloaded to the user's computer. The malware looks for vulnerable versions of Flash and uses it to gain control of the infected computer. This is only one example of malware targeting the general public.

SLEMBUNK, belongs to a family of Android trojan applications, targets mobile banking applications [374, 375]. Users become infected by downloading common popular applications that are infected. It attempts to steal log-in credentials of the mobile banking applications by detecting the launch of legitimate applications and displaying fake log-in interfaces. This version of malware has targeted over 33 mobile banking applications and covered North America, Europe, and Asia Pacific [375].

Duqu, Stuxnet, Flame, Red October, APT28, and SLEMBUNK have caused a renewed interest in cyber-security. This is partly due to the high consequences of a successful attack as well as the weaknesses that continue to exist in networks. As we increase our understanding of the weaknesses, vulnerabilities, and interactions concerning networks, we gain a better understanding of how to protect and defend critical networks with limited resources.

1.1.3 The New Battlefield.

These recent attacks have transitioned the U.S. military from its traditional view of a battlefield of a physical space into the new battlefield of cyberspace. First it is necessary to define this new battlefield:

6

Definition 1. *Cyber* - *"Of, relating to, or characteristic of the culture of computers, information technology, and virtual reality." [86]*

Definition 2. *Cyberspace* - *Is "an operational domain whose distinctive and unique character is framed by the use of electronics and the electromagnetic spectrum to create, store, modify, exchange, and exploit information via interconnected Information-Communication Technology (ICT) based systems and their associated infrastructures." [207]*

Definition 3. *Cyberattack* - *Combing computer network attack and defense methods to use as an individual act in order to cause damage, destruction, or casualties for personal gain or a limited cause [221].*

Definition 4. *Cyberwarfare* - *Combining computer network attack and defense methods with special technical operations by states or political groups, in order to cause damage, destruction, or casualties for political effect [85] [221].*

Definition 5. *Cyber Physical Systems (CPS)* - *"Are integrations of computation, networking, and physical processes. Embedded computers and networks monitor and control the physical processes, with feedback loops where physical processes affect computations and vice versa" [295]*

This new battlefield is defined within the increased dependence/reliance on electronic systems. More and more of these systems are cyber-physical systems - systems making life and death decisions as these appear in cars, planes, buildings, phones, and medical devices.

Cyber may also be used as a weapon which requires some of the same principles as kinetic warfare to be considered as well as additional principles specific to cyberspace. Principles that must be considered include: lack of physical limitations, kinetic effects, stealth, mutability and inconsistency, identity and privileges, dual use, infrastructure

control, information as operational environment, assured response, and escalation control. The utility of cyberwarfare has benefits and disadvantages, with tradeoffs being made as different strategies are selected and the adversary reacts.

Game theory has been applied to model and analyze cyber security issues and conflicts [72, 165, 306, 326, 331]. Game theoretic applications have often focused on symmectric games where players did not have distinct advantages (known or unknown) over other players. These advantages often happen in military engagements as opposing forces try to protect information. Hypergame theory can model and help understand the strategies that yield the greatest utility during cyber warfare while considering advantages/disadvantages. That can help to maximize the benefits while minimizing the disadvantages. It can also model the opponent in order to better understand any response such as "assured responses" and "escalation control."

1.2 Problem Statement

As the global economy develops, manufacturing is no longer the only means of power and influence between countries. Instead manufacturing is becoming concentrated among a handful of countries where goods can be produced at record low prices. As countries like the U.S. watch manufacturing disappear, another commodity is developing - knowledge as capital. Knowledge as capital can be in the form of military weapon technology, cutting edge medical technology, and even foreign intelligence. This capital can lead to increased power over other countries with less means or countries in need of help defending themselves. The disadvantage is that most of this knowledge is stored electrically on networked computer systems and is susceptible to attack.

Over time this knowledge store has become the target of adversaries, who find it cost effective to steal it instead of producing it themselves. Numerous cyber attacks against U.S. contractors have shown the importance placed on this information by

8

adversaries [1, 8, 142, 237, 249]. These cyber attacks against critical national security systems are showing no signs of ceasing. Photos of China's new fighter have emerged that appear the technology used in the aircraft is very similar to the U.S.'s F-22 Raptor [197]. As shown in Figure 1.1, the technology in both aircraft is strikingly similar, an indication of industrial espionage and cyber theft.

Figure 1.1: U.S. media comparison of U.S. F-22 and China J-20 [197].

The United States Air Force, as well as the rest of the Department of Defense (DoD), has an interest to build stronger defenses to cyber attacks. Building these defenses require modeling, of the cyber attack as a conflict iof the adversary of the true state of the U.S. network, and understanding of the interactions of all of these elements over time. These interactions are often complex and not fully understood. This incomplete information leads to changing models or models full of assumptions which are hard to apply to general attacks/conflicts.

1.3 Temporal Hypergame Approach

While hypergames provide a clear and concise method of displaying information about a scenario for analysis, this research proposes to address three current issues

when applying hypergame theory to complex domains by extending hypergame theory with temporal concepts to improve the representation of the cyber-conflict. In order to achieve this goal, there are two objectives:

1. Develop a temporal hypergame mathematical framework and define trust, distrust, and deception formally.

2. Extend the theoretical application of hypergames to cyber related conflicts.

A formal framework for a temporal hypergame is developed in order to allow hypergames to model temporal aspects of games that are currently neglected. It incorporates temporal logic to ensure decision makers are able to build a better model of the problems decision makers face. Given the unbalanced nature of hypergames where one player may not be aware of all the possible actions and outcomes in a given game, deception is an important concept. While past literature discusses the usefulness of using hypergames to model deception, no formal mathematical model has been presented. This research takes key concepts from the literature and presents a formal mathematical model over the developed framework. In order to analyze the framework, it is applied to a complex cyber attacker/defender conflict of a SCADA network.

The temporal framework for hypergames continues to incorporate the knowledge of the domain experts, that requires human-in-the-loop interactions. Thus, this methodology for analyzing conflicts/decision making is not intended to be totally automated. While an automated system could be created, it is not the focus of this research.

1.4 Significance

The alarming rate at which cyber threats continue to evolve has demonstrated a need to help decision makers model the threat, select the best strategy based

on resources, information, and beliefs about the adversary. The loss of critical information or systems to the malicious actions of the adversary can have far-reaching consequences and jeopardize the success of the mission. The deployment of accurate and timely threat deterrents, damage reduction techniques, and intelligence gathering techniques can prevent or lower the cost of recovering from a cyber attack.

This research provides the U.S. Air Force, Department of Homeland Security, and other government agencies a method to organize information, resources, and beliefs in order to reason about the cyber battle space and make rational decisions. The extension of hypergame theory with temporal constructs allows the model to evolve over time leading to a more actuate model for decision makers. The temporal hypergame framework developed allows for a formal definition of deception over the model. This helps improve the ability to identify and limit deception in decision making.

1.5 Document Organization

This document is organized as follows. Chapter 2, 3, and 4 provide a review of the background literature. In particular, relevant literature about game theory and decision theory is provided in Chapter 2; the differences between game theory and decision theory are also discussed. Chapter 3 covers the literature on hypergame theory and representations, such as HNF, and develops the foundation of using hypergames to model complex conflicts/decision making processes. A temporal logic overview is presented in Chapter 4, providing a concise model to capture temporal modeling aspects.

Chapter 5 presents mathematical models for the two hypergame models presented in the literature. The models include the original model developed by Bennett and the improved representation developed by Vane. These mathematical models serve as the foundation on which to begin integration of temporal constructs for HGT.

Chapter 6 integrates the temporal aspects 4 with the Hypergame Theory presented in Chapter 3 and the hypergame models presented in Chapter 5. The enhancements to the temporal model and the adaptability of hypergames are discussed, followed by the entire temporal hypergame framework being presented. The temporal hypergame framework is used to define key concepts important to hypergames. The first is the definition of trust (as well as distrust), using the temporal constructs of the framework. From the definition of trust, the concept of misperception and deception is defined and discussed. Theorems about repeated games are discussed in terms of applicability to the temporal hypergame model.

Chapter 7 applies the temporal framework. The first two applications are classical game theory examples, the Prisoner's Dilemma and the game of Chicken. These games show the framework is able to represent simple classical games (for understanding) and capture the iterated (temporal) nature of the games. The framework is applied to a SCADA cyber security hypergame. This application exercises the framework and provides a clear application to a complex hypergame.

Chapter 8 states the research conclusions and summarizes possible future work. The conclusions contain an organized list of findings and definitions evolved from the framework of this research.

II.   Tale of Two Theories

Game theory is a bag of analytical tools designed to help understand the phenomena that observed when decision-makers interact [15]. Decision theory is a formal mathematical theory about how decision makers make rational decisions as they interact with their environment. What is the difference between game theory and decision theory? There is a division between decision theory where the outcome depends on the players decisions and the impersonal universe, and game theory depends on the decisions made by interacting with other players. This chapter discussed game theory and decision theory, as well as the differences between the two theories.

2.1   Game Theory

Game theory is based on rationality and utility theory. Often it is assumed that human beings are rational and always seek the best alternative when presented with a set of possible choices. This increases the chance of predictability by narrowing the range of possibilities. Utility theory is based on rationality and that an agent will always maximize their utility through their choices. Utility is a quantification of a person's preferences with respect to certain objects and the environment. Game theory is highly mathematical and assumes all interactions can be understood and navigated by presumptions.

Game theory asks two questions about the interaction of the decision-makers [364]:

- How do individuals behave in strategic situations?

- How should these individuals behave?

Answers to the two questions do not always coincide [1].

2.1.1 Definitions.

The basics of game theory are presented in detail for greater clarity when the concepts are applied in following sections. The following set of definitions provides a basic understanding of game theory terminology. This terminology is further developed in [270], [88], [248], or [258].

- **Game** A set of interactions between players, where constraints and utility are considered without concern for response from other players.

- **Player** A decision maker in a game. The decision maker chooses actions in order to carryout a strategy in a game. A player may be a person, as well as an animal, machine, or group of people.

- **Action** A valid move in a game.

- **Payoff** Quantitative measurement of the reward received by each player at the end of a game.

- **Strategy** The set of actions an individual player can make in a game.

- **Pure Strategy** A strategy that a player follows in every attainable situation during a game.

- **Mixed Strategy** A strategy that consists of a multiple set of actions that are chosen based on a probability distribution that determines how often each action is played.

- **Dominating Strategy** A player's strategy is said to dominate the strategy of another player if it always results in a better payoff regardless of the strategy of the other player. The strategy weakly dominates the other strategy if it is always at least as good.

14

- **Rational Behavior** Each player has a consistent set of payoffs for possible outcomes and chooses the strategy that maximizes the player's payoff.

- **Perfect Information** Information concerning an opponent's past moves are well known in advance. Tic-tac-toe, chess, checkers, and go are examples of games with perfect information.

- **Imperfect Information** Partial or no information concerning the opponent's past moves are given in advance of the player's decision. Imperfect information may be diminished over time if the same game is repeated with the same opponent.

- **Complete Information** All of the players in a game knows the strategies and payoffs of every player. Complete information is not the same as perfect information, because the former does not consider the actions each player have taken in the past.

- **Incomplete Information** Partial or no information concerning the opponent's strategies or payoffs are given in advance of the player's decision.

- **Signaling** Strategies that use signals.

- **Signals** Objective evidence or actions in a game that offer credible proof of a player's information.

- **Screening** A player can create a scenario in a game where another player must take some action that reveals credible proof of that player's information.

- **Screening Devices** Strategies that use screening.

- **Normal Form** A matrix representation of the possible outcomes based on each decision by the players.

- **Nash Equilibrium** A set of actions in which neither player can increase their utility by unilaterally changing his or her strategy. If a player uses mixed strategies, then the expected value of the payoffs are maximized.

- **Simultaneous Game** Each player chooses an action simultaneously without knowing which action was chosen by the opponent. Each action may happen at different times but the actions are unknown to each player. A one-shot simultaneous game is also called a static/strategic game.

- **Sequential Game** Each player chooses an action in a predetermined order. Players are allowed to observe the decisions of the other players before making a decision.

- **Non-cooperative Game** Players in the game are in conflict with another player. There is no incentive for players in conflict to compromise.

- **Zero-sum Game** The sum of payoffs remains constant during the course of the game. Being well informed always helps the two players in conflict.

- **Bayesian Game** A player assigns a "type" to all of the other players at the start of a game. The information about the other player's strategies and payoff is incomplete. The outcome of the game is predicted using Bayesian analysis.

- **Dynamic Game** Players consider their actions in multiple stages of a game. It is the sequential structure of the decision making by players in a strategic game. The sequences are either finite or infinite.

- **Stochastic Game** Involves probabilistic transitions through states of the game. The game has a start state where the players choose actions and receive a payoff based on the current state. The game transitions into a new state using a probability from the players actions and the current game state.

2.1.2 Overview.

Game theory can be traced to Talmud (0-500 AD) with results similar to modern game theory [21]. Cournot [79], Edgeworth [92], and Bertrand [49] published the first papers on oligopoly pricing and production. These original papers were considered special models. Modern game theory was born with the collective work of Zermelo [372], Borel [56], and von Neumann [257] (original non-english [256]). In 1944, von Neumann and Morgenstern published a seminal book on zero-sum cooperative games where players form coalitions [258]. Maximizing expected utility payoff is first attributed to Bernoulli [48]; the modern idea is from von Neumann and Morgenstern [258]. von Neumann and Morgenstern showed that for each rational player there is a way to assign utility numbers to the possible game outcomes, in which the player will choose the outcome that maximizes the player's expected utility.

Zermelo [372] applied game theory to the game of chess. While the initial theory states that a game cannot end in a draw and one player has a winning strategy if: 1) the game is finite, two-person; 2) perfect information; 3) chance does not affect the decision making process. Zermelo's theorem has been generalized for game theory [234] [316]:

Theorem 1. *Every finite game of perfect information has a pure strategy Nash equilibrium that can be derived by backward induction. Moreover, if no player has the same payoffs at any two terminal nodes, then there is a unique Nash equilibrium that can be derived in this manner.*

von Neumann [256] developed the Minimax Theorem in 1928. It is the fundamental theorem of game theory that states a game has optimal mixed strategies (for finite, zero-sum, two-person games). The Minimax Theorem is formalized by [315, 365]:

Theorem 2. *Formally, let X and Y be mixed strategies for players A and B respectively. Let M be the payoff matrix. Then*

$$max_X min_Y X^T MY = min_Y max_X X^T MY = v$$

where v is called the value of the game and X and Y are called the solutions.

In the early 1950's Nash contributed to noncorporative and corporative game theory [251–253]. Nash [250] builds on von Neumann and Morgensterns work by assuming the absence of coalitions where each player acts independently. His work proves for each finite non-cooperative game there is at least one equilibrium point assuming the players are rational. A Nash equilibrium is a strategy where none of the players can improve their payoff by unilaterally changing their strategy. In a game of mixed strategies, every game will have at least one Nash equilibrium.

Definition 6. *Nash equilibirum* *[252] - A strategy pair (p, q) is a Nash equilibrium of a game G if given all other strategies, r: Player1(G(p,q)) ≥ Player1(G(r,q)) and Player2(G(p,q)) ≥ Player2(G(r,q))*

Nash also showed that in a game with mixed strategies, and not just pure strategies every game will have at least one Nash equilibrium [250]. The Nash Existence Theorem stated formally:

Theorem 3. *Every finite game has a mixed strategy Nash equilibrium.*

Additional work in game theory was completed during World War II (WWII) by the Rand Corporation. This research combined military planning with game theory research, leading to developments in reasoning under uncertainty.

Selten [320] presented the idea that in games where all the players can choose contingent plans not all of the Nash equilibria are equally reasonable. This is due to the fact that players can make empty threats - contingent plans - that may or may

not be carried out. The concept resulted in subgame perfection to eliminate equilibria that depend on threats.

When describing a game model, two forms can be used. The extensive form using a game tree or graph to model the possible strategies and all possible outcomes. As shown in Figure 2.1, there are two players, player 1 (P1) and player 2 (P2). Player 1 chooses first by selecting strategy U or strategy D. Player 2 is allowed to see the action of Player 1 and then can choose strategy L or strategy R. After this, players receive the outcomes located on a terminating branch on the tree. The extensive form is used in [15, 325]. The strategic form (or normal form) uses a matrix to represent the players, their strategies, and the possible outcomes as shown in Figure 2.2. While the majority use the strategic model [23, 72], there are a few that use both models [23, 162]. The strategic form has a simplicity that lends to a straightforward analysis. For cyber security games players are not making decisions one at a time in sequence, as in the extensive form game. Instead players are making decisions dynamically and possible at the same time, where the strategic form is better suited.

2.1.3   Classical Games.

The following section discusses some of the games used in classical game theory. A brief overview is given in Table 2.1 of the game characteristics. These are a few of the important games and not an inclusive list. A more detailed list can be found on Wikipedia [77].

2.1.3.1   Prisoner's Dilemma.

In game theory, the Prisoner's Dilemma described by Melvin Dresher [90], Merrill Flood [104], and Albert Tucker [284] is a classic example of how the interaction between two individuals leads to cooperation or not. In the Prisoner's Dilemma there are two players, Prisoner A and Prisoner B. Each player can choose one of two actions,
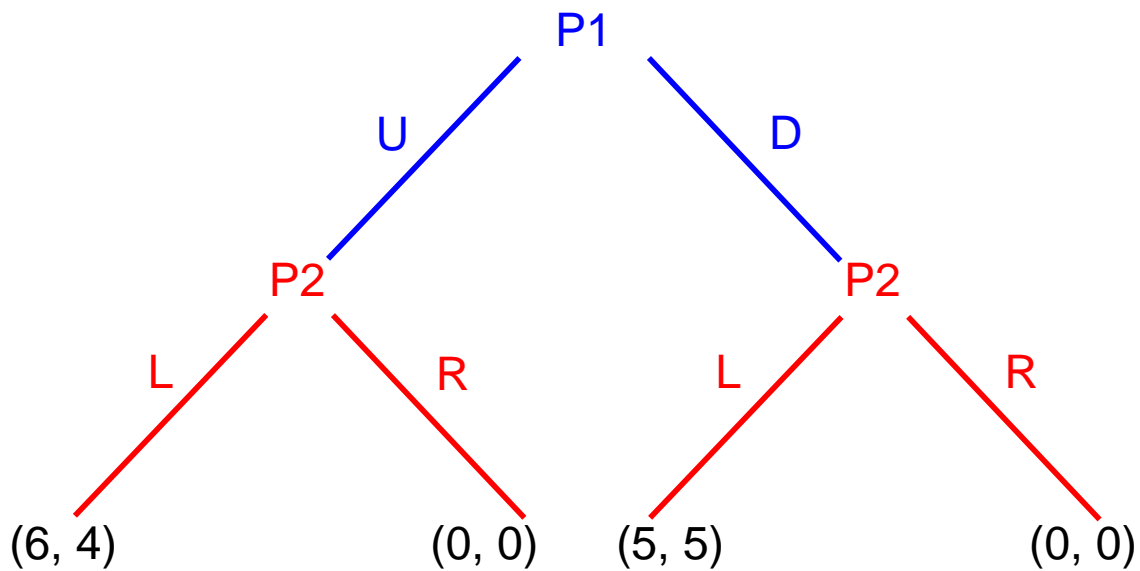
Figure 2.1: Example of Game in Extensive Form.

Figure 2.2: Example of Game in Strategic (Normal) Form.

Table 2.1: Overview of Games in Game Theory.

Property/ Game	Prisoner's Dilemma	Chicken (Hawk-Dove)	Matching Pennies	Rock, Paper, Scissors	Battle of the Sexes
Players	2	2	2	2	2
Strategies	2	2	2	3	2
# of Pure NE	1	2	0	0	2
Sequential	No	No	No	No	No
Perfect Information	No	No	No	No	No
Zero-sum	No	No	Yes	Yes	No

either the player can choose to cooperate by staying silent or defect by betraying the other player. The consequence of selecting an action results in no jail time (jail(0)) or jail time (jail(x) where x is the amount of time in jail). The payoff function is if the player cooperates and the other player cooperates, both receive one year in jail, otherwise if the other player defects the player receives 10 years in jail while the other player receives no jail time. If both players defect, then both players receive five years in jail. This game is shown in normal form in Figure 2.3.

In the Prisoner's Dilemma, betrayal always has a higher reward than cooperation. If all players are purely rational, then they would betray each other by choosing to defect. It is clear both players would receive a better reward by both cooperating.

2.1.3.2   Chicken.

Chicken , also known as the Hawk-Dove or Snow-Drift game, is a game where players prefer to not yield to each other, while the worst outcome is obtained when both players fail to yield. The game of Chicken and the Hawk-Dove game are identical

Figure 2.3: The Prisoner's Dilemma - Normal Form.

from a game theoretic view. The differences in names are from each game being used in different research areas such as economics or biology [270].

In Chicken [293] two players drive towards each other on a possible collision course. Each player has two options: swerve or continue straight. It is possible that if neither player swerves, both may die in a head-on collision. If a player swerves they are called a "chicken", which is considered bad. Since the worst outcome for both players is a collision, it is presumed that the best outcome is for each player to stay straight while the other player swerves. Here each player risks the most, while attempting to secure the best outcome. This is represented in Figure 2.4, where the benefit of winning is 1, the cost of losing is -1, and crashing costs -10.

The cost of swerving is trivial compared to the cost of a crash, it is therefore reasonable to assume the strategy to swerve is likely. But if the player's opponent is considered reasonable, then it may be better to stay straight, believing the other

Figure 2.4: The Game of Chicken - Normal Form.

player will be reasonable and swerve to avoid the collision. In this game of chicken, the pure strategy equilbria are the two outcomes where one player stays straight and the other swerves.

2.1.3.3 Hawk Dove.

The name Hawk-Dove comes from biological literature by John Maynard Smith [286] and the traditional payoff matrix is given in [335] and [334]. It is similar to the game of Chicken.

2.1.3.4 Differences in the Prisoner's Dilemma and Chicken.

In the Prisoner's Dilemma, both players have a dominating strategy. This means that regardless of the strategy chosen by the opponent, the player should choose a specific action. In this case the player would choose to defect. If both players choose their dominating strategy, then the outcome is a Nash equilibrium. In the sense of a Pareto equilibrium this is inefficient because all of the players would prefer a different

outcome. While each player has a preferred strategy, collectively the strategies result in an inferior outcome.

In Chicken, none of the players have a dominating strategy. The best strategy is not cooperate with the other player; therefore, swerve if the other player stays straight or stay straight if the other player swerves. This position is adversarial as each player has a preferred strategy, but are in a rivalry with each other.

2.1.3.5 Matching Pennies.

A simple game used in game theory is Matching Pennies [244] [265], which is equivalent to the game Odds or Evens and is the two strategy equivalent of Rock, Paper, Scissors. In the game, two children try to determine who is required to do the nightly chores. The children first determine who is "same" and the other is "different". Then each child places a penny face up or face down in their palm concealing it from the other player. The children reveal both coins simultaneously; if both coins match (both coins are heads or both are tails) then the "same" child wins, otherwise if they are different (one coin is heads and the other is tails) then the "different" child wins. Figure 2.5 shows the normal form for the game of Matching Pennies.

The game of Matching Pennies is a zero-sum game as shown in Figure 2.5. Given the pure strategies, there is no set of pure strategies where both players would switch strategies Therefore the equilibrium is obtained by playing mixed strategies. Each strategy has an equal probability of being played, resulting in each player receiving an expected payoff of zero. The mixed strategy makes the opponent indifferent to playing pure strategies, so neither player has incentive to switch to another strategy.

2.1.3.6 Battle of the Sexes.

Another classical game in game theory is the Battle of the Sexes, or Bach or Stravinsky [270], a coordination game between two-players. In the game a husband and wife have agreed to meet for an evening together, but neither can remember if

Figure 2.5: The Game of Matching Pennies - Normal Form.

they decided to attend an opera performance or a football game. It is considered common knowledge that both will forget. The husband prefers to attend the football game, while the wife prefers to attend an opera performance. Both the husband and the wife prefer to be at the same event instead of different events. The goal of the game is to determine where each should go assuming they cannot communicate. Figure 2.6 shows an example payoff matrix for this game.

The Battle of the Sexes has two pure strategy Nash equilibria, where the couple either attend the opera or a football game. A mixed strategy Nash equilibrium exists in both games, where the players choose to attend their preferred event more than the other. This means the Nash equilibria are deficient in some way. In the pure strategy Nash equilibria, one player will consistently do better than the other, while the mixed strategy Nash equilibrium will cause the players to appear at different events the majority of the time.

	Husband	
	Opera	**Football**
Opera	(3, 2)	(1, 1)
Football	(0, 0)	(2, 3)

Figure 2.6: The Game of the Battle of the Sexes - Normal Form.

2.1.4 Beyond Normal and Extensive Forms.

There are game theoretic representations for player interactions other than normal and extensive forms. These other representations provide important models for representing realistic interactions, where the games may not be finite and instead repeated with no end or that the set of agents is uncountably infinite. Repeated Games, Bayesian Games, and Stochastic Games provide richer models with which to analyze player interactions.

2.1.4.1 Repeated Games.

In real-life strategic situations, players interact over time often repeating the same interaction. This happens where trust and social pressure exists between multiple parties, such as trades without a legal contract. Chamberlin describes a repeated game where oligopolists my collude on higher prices [69]. In 1963, Macaulay observed the relationship between a business and its suppliers are largely based on reputation and the threat of losing future business [228].

Repeated games do not allow for the set of actions available to the players or the payoff functions to change according to the past play of the players[115]. It is possible for the action taken by one player to open up new actions for opponents in real-life, for example war where mass killings can change public option. Payoffs can also change overtime in real-life, for example availability of computer network resources may not be devastating for short outages while long term outages are more costly. This limits the ability of repeated games to model phenomena such as business investment in capital or learning about the physical environment [115].

A repeated game is a situation where the same game is played multiple times consecutively by the same players. This repetition allows for the possibility the players will utilize cooperative strategies during interaction that are not available in one-shot, or non-repeated, games. The specific game being repeated is called a stage game (i.e., each time a game is played it is a stage in the repeated game). The Prisoner's Dilemma is shown in Figure 2.7 as a two-round, finitely-repeated game in normal form and in Figure 2.8 in extensive form.

Figure 2.7: Two-Round, Prisoner's Dilemma in Normal Form.
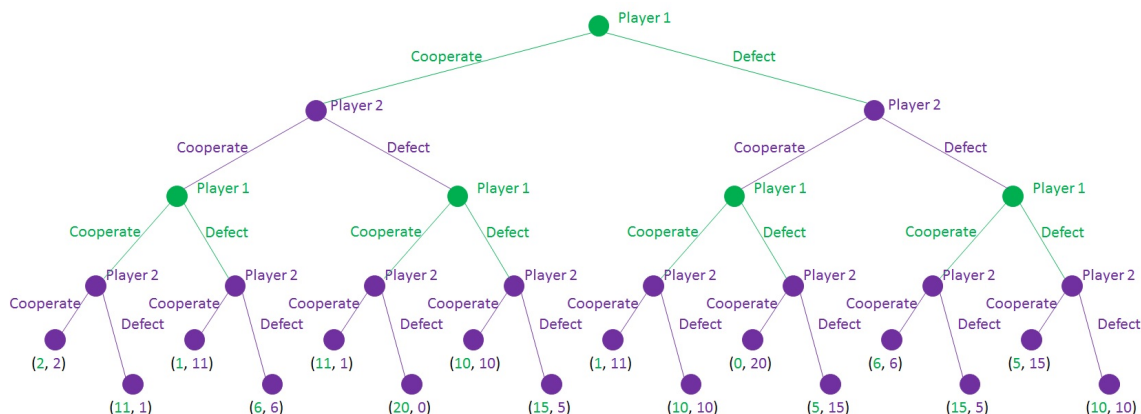
Figure 2.8: Two-Round, Prisoner's Dilemma in Extensive Form.

The normal form representation of a repeated game, similar to Figure 2.7, hides key components of the game. These key components are important and are [327]:

- Do the other players know what their opponent(s) did earlier?

- How much to the players remember of the past?

- What is the utility of the entire repeated game?

These key components are approached from two different types of games: finitely repeated, or finite-horizon, games where the game is repeated but eventually ends and infinitely repeated, infinite-horizon, games where the game is repeated but indefinitely.

In repeated games, players have the ability to cooperate from round to round. The key to cooperation is that the players must have incentive to follow through on the commitments they have made to other opponents. The multiple round Prisoner's Dilemma is an example of the ability of players to cooperate in a finite game.

In the Prisoner's Dilemma, for any number of rounds greater than one, the dominant strategy in the last round is to defect no matter what happened in the past.

The dominant strategy is common knowledge among players, so the players cannot affect the outcomes of the last round. Thus in the second-to-last round the dominant strategy is to defect. Using induction (backwards-induction) the only equilibrium in the finite repeated Prisoner's Dilemma is to defect since none of the players can make a credible promise of cooperation.

This allows the outcome of a finitely repeated game to be determined by analyzing a one-shot version of the same game. Backward induction will always lead to the same subgame perfect Nash equilibrium (SPNE). It is not always the case a finite game will empirically lead to backward induction, while backward induction in a finite game is logically correct assuming all players are rational.

In 1978, Reinhard Selten proposed the chain store paradox [322]. He proposed a finitely repeated game with two players, where the incumbent firm is a monopolist with a chain of stores in twenty different locations. At each location, the chain store is challenged by a rival firm. This game is a sequential game, in which the first firm decides whether to enter or not at the first location then the chain store must decide to fight or accommodate. Play is continued with the next firm deciding, etc. Using backward induction, the chain store will accommodate in the last round and will therefore accommodate in every round of the game. Selten calls this the "induction hypothesis" and shows the chain store can reach a better outcome by fighting the first fifteen rivals and accommodating the last five.

If a game is repeated, as long the number of repetitions is finite, then there is an unique SPNE. This is stated formally in Theorem 4. If a stage game has multiple Nash equilibria, then the strategies can be history-dependent. This results in a possible SPNE in the repeated game, where for some repetitions, actions are played that are not part of the Nash equilibria of the stage game [193].

Theorem 4. *Suppose $0 \leq T < \infty$ and the stage game has a unique (possibly mixed) Nash equilibria, α^*. Then the unique SPNE of the repeated game is the history-independent strategy profile σ^* s.t. $\forall \ t \in T$ and h^t, $\sigma^*(h^t) = \alpha^*$. Where T is the set of game iterations or rounds numbered 1, 2, ..., n.*

Payoffs in infinitely repeated games cannot be precalculated. When the infinitely repeated game is modeled using extensive form and an infinite tree. There is no way to attached the payoffs to any terminal nodes (as none exist in the infinite tree), or can the payoffs be the sum of the individual stage games. The two most common way to calculate payoffs in an infinitely repeated game are average reward and discounted reward [327]:

- Average Reward - Let $(r_i^{(1)}, r_i^{(2)}, ...)$ be an infinite sequence of payoffs for the player i, then the average reward for player i is:

$$\lim_{k \to \infty} \frac{\sum_{j=1}^{k} r_i^{(j)}}{k}$$

- Discounted Reward - Let $(r_i^{(1)}, r_i^{(2)}, ...)$ be an infinite sequence of payoffs for the player i, and β be a discount factor with $0 \leq \beta \leq 1$, then the discounted reward for player i is:

$$\sum_{j=1}^{\infty} \beta^j r_i^{(j)}$$

The sequential nature of repeated games allows players to adapt strategies which depend on the actions chosen in the preceding games; these strategies are called contingent strategies. Trigger strategies are contingent strategies, where a player plays cooperatively as long as the opponents do as the same but any change by the opponent will trigger a period of punishment. Two well-known trigger strategies are the grim strategy and tit-for-tat [88]. In the grim strategy, the player cooperates with their opponents until the opponent defects at which time the player will defect for

the rest of the game as punishment. In the tit-for-tat strategy, the player plays the same strategy as their opponent did in the previous round of the game. This means the punishment only lasts as long as the opponent chooses to not cooperate.

The strategy space of an infinitely repeated game is large which makes it hard to characterize all the Nash equilibria of the game. The Folk Theorem [112] does not characterize the equilibrium strategy profiles in a game, but it does characterize the payoffs obtained from the strategies. It states that the average rewards obtained when in equilibrium are the same as the rewards obtained under mixed strategies in a single-stage game, where each player receives a payoff of at least what he would receive if opponents played minmax strategies [327].

Let a repeated infinitely game, $G = (N, A, u)$ and $r = (r_1, r_2, ... \ r_n)$ be the strategy profile, where N is a set of players, A a set of actions, and u a set of payoffs. Then player i's minmax value, the utility received with opponents play minmax strategies and player i plays his best response, is v_i.

$$v_i = min_{s_{-i} \in S_{-i}} max_{s_i \in S_i} u_i(s_{-i}, s_i)$$

Given a strategy profile $r = (r_1, r_2, ... \ r_n)$:

- Enforceable - The payoff profile is enforceable if $\forall$ i $\in$ N, $r_i \geq v_i$

- Feasible - The payoff profile is feasible if there exists rational, non-negative values α_a, such that $\forall i$, $r_i = \sum_{a \in A} \alpha_a u_i(a)$, with $\sum_{a \in A} \alpha_a = 1$

The Folk Theorem states that for any game, G, and any payoff profile, r:

1 If r is the payoff profile for any Nash equilibrium s of the infinitely repeated G with average rewards, then for each player i, r_i is enforceable.

2 If r is both feasible and enforceable, then r is the payoff profile for some Nash equilibrium of the infinitely repeated G with average rewards.

The Folk Theorem stated formally [272]:

Theorem 5. *Let G be a finite, simultaneous move game of complete information, let $(u_1^*, ..., u_n^*)$ denote the payoffs from a Nash equilibrium of G, and let $(u_1, ..., u_n)$ be a feasible payoff of G. If $u_i \leq e_i \ \forall \ i \in N$ (the set of players) and if γ is sufficiently close to 1, then there exists a Subgame-perfect equilibrium (SPE) of the infinitely repeated game $G(\gamma)$ that achieves an average payoff arbitrarily close to $(u_1, ..., u_n)$.*

In repeated games, the one-shot deviation principle states that for any player when profitable deviations from a SPNE are considered, only the strategies where the player plays as he is expected to at all round except one (i.e. at a single history the player behaves differently) [193].

Theorem 6. *Fix a strategy profile σ. A profitable one-shot deviation for player i is a strategy $\sigma_i' \neq \sigma_i$ s.t.*

- *there is an unique history $h^{t'}$ such that $\forall \ \hat{h}^t \neq \hat{h}^{t'}$, $\sigma_i'(\hat{h}^t) = \sigma_i(\hat{h}^t)$.*

- *$u_i(\sigma_i'|h^{t'}, \sigma_{-i}|h^{t'}) > u_i(\sigma_i|h^{t'})$*

  Where $\sigma_i|h^t$ is the restriction of strategy σ_i to the subgame following history h^t.

In Theorem 6, the requirement states there is only one history at which the strategies are different. The differences in strategies can have a significant effect on the path of play, since all histories after $\hat{h}^{t'}$ may depend on what was played in $\hat{h}^{t'}$. The second requirement states the deviation has to be profitable. The profitability of the deviation is defined as a conditional on history h^t being reached, since strategy σ may not lead to h^t being reached. By definition this means a Nash equilibrium can have a profitable deviation, but this cannot be the case so the following lemma results:

Lemma 7. *A strategy σ is a SPNE iff there are no profitable one-shot deviations.*

The one-shot deviation principle implies repetition of the stage-game Nash equilibrium is a SPNE of a repeated game [193]. Formally this is stated as:

Theorem 8. *A strategy profile σ is history-independent if $\forall\, h^t$ and $\hat{h}^t$, $\sigma(h^t) = \sigma(\hat{h}^t)$*

Theorem 8 implies the existence of an SPNE in infinitely repeated games.

Lemma 9. *If the stage game has a Nash equilibrium, then the repeated game has a SPNE.*

2.1.4.2 Stochastic Games.

Stochastic games where first introduced by Lloyd Shapley in 1953 [324]. In stochastic games, the same stage game is not always repeated. Stochastic games generalize both repeated games and Markov decision processes (MDPs). A repeated game is a stochastic game where only one stage game is repeated, while an MDP is a stochastic game with only one player [327].

A stochastic game is a repeated game where agents play games from a set of games repeatedly. At any iteration in the future, the game played depends only on the previous game and the actions chosen by the players in that previous game. Stochastic games are a generalization of the Markov decision process. This is defined by multiple players, one reward function per player, and the action chosen by both players determines the transition and reward functions. A stochastic game is modeled as [327]:

S = (Q, N, A, P, r)

- Q is a finite set of stage games

- N is a finite set of players

- A = $A_1 \times ... \times A_n$ where A_i is a finite set of actions for player i

– P : Q × A × Q → [0,1] is the transition probability, P(q, a, $\hat{q}$) is the probability of transitioning form state q to state $\hat{q}$ after action profile a

– R = r_1, ..., r_n where r_i : Q × A → $\mathbb{R}$ is a payoff function for player i.

In stochastic games it is assumed the strategy space of the agents is the same in all games in the set of stage games. Resulting in the only difference between any game being the payoff function. This assumption can be removed without adversely affecting the overall stochastic game [327]. Stochastic games are also often modeled with finite state space and action sets (as shown in the previous stochastic model), which are not needed to receive the benefit from stochastic games and can be relaxed [241].

There are three types of strategies of interest over the strategy space of an agent in stochastic games. A strategy space is defined as $\prod_{t,H_t} A_i$, where $h_t = q^0$, a^0, q^1, a^1,....,a^{t-1}, q^t) denotes the history of t stages and H_t is the set of all possible histories. Given an agent's strategy can consist of any mixture over deterministic strategies:

- Behavioral Strategy

    – is where the mixing of strategies takes place at each history independently, instead of only once at the start of the game

    – $s_i(h_t, a_{i_j})$ gives the probability of playing action a_{i_j} for the history h_t

- Markov Strategy

    – is where for time t, the action probability distribution only depends on the current state

    – $s_i(h_t, a_{i_j}) = s_i(h_t\prime, a_{i_j})$ if $q_t = q_t\prime$, where q_t and $q_t\prime$ are the final states of h_t and $h_t\prime$

- Stationary Strategy

– is where the strategy has no dependence, not even on time

– $s_i(h_{t_1}, a_{i_j}) = s_i(h_{t_2}', a_{i_j})$ if $q_{t_1} = q_{t_2}'$, where q_{t_1} and q_{t_2}' are the final states of h_{t_1} and h_{t_2}'

For the discounted-reward payoff case, a Nash equilibrium exists in every stochastic game. If the strategy profile only consists of Markov strategies then it is called a Markov perfect equilibrium (MPE), and is a Nash equilibrium regardless of the starting state of the game. MPE is similar to the subgame-perfect equilibrium in perfect information games [327].

2.1.5 Applications to Cyber Security.

The large number of network intrusion incidents, especially Stuxnet, Flame, and Duqu are causing a rapid evolution of malware and defense techniques. As the authors in [331] discuss the evolution of these techniques is similar to a game between malware authors and security analysts, with each trying to win the game by outperforming the opponent. This shows cyber attack and defenses can be modeled as a game between multiple players. Insight into the best strategy choices can be obtained by using a game to model the adversarial players. In most game theory models for cyber security, the objective is to select the security measures with the lowest cost, while achieving the highest level of security [326]. This shows game theoretic models can be used to guide decisions for network defense.

There is an inverse relationship between computer/network usability and computer/network security - a double edge sword. When a system is more usable, it becomes a liability for the operator, while a system with higher security results in decreased usability for authorized users. When authorized users are unable to use the system for authorized tasks, this is the same as a denial of service attack by a potential attacker. It is necessary to realize the need to carefully balance usability and security so authorized users are not hampered by security and give up, leading to

the cyber-battle being lost [6]. Sometimes it is necessary to conduct trade-off analysis to determine a balance between usability and security, while protecting critical data and resources and minimizing the impact on authorized users.

Human administrators are able to reason and make decisions on how to use defensive capabilities to minimize threats and allow users to maximize their usability experience, but may overlook certain combinations of defenses [306]. Computers can enhance the decision making of human administrators by analyzing large amounts of data and finding all possible combinations of defensive capabilities, allowing exceptions to be found and optimization to take place. One should not infer a human in the loop is not needed, instead this shows the complexity of the problem and how computers can aid in finding a solution. When using game theory, it is necessary for the model to be as accurate as possible in order to achieve the best possible solution to the real-world situation.

Scalability of a game model can be problematic as the number of variables can grow exponentially. This leads to most models using a simplification of the real-world problem in order to reduce the variables and the complexity of the model. One simplification most models use is to reduce the game to a two player game, such as one attacker and one defender. Another simplification is to reduce the number of strategies in the game model, because for each additional strategy in the player's strategy set the number of outcomes increases by the total of the other players' strategies. Stability concerns lead to models with two players, each with two strategies. For example, in [72] there is an attacker and a defender, while the attacker can attack or not attack and the defender can defend or not defend.

2.1.6   Related Work to Cyber Security.

Previous work has built information warfare models using game theory. David Burke [62] presents an information warfare model, that is a repeated non-zero sum

game with incomplete information. The model consists of two players - an attacker and a defender. Defenders try to protect the most valuable information while attackers try to obtain highly valuable information. Each player is one of three types: social, infrastructure, and node. Each type of player prefers one type of information to all others (this player realizes higher payoffs for defending/obtaining this information). The social player prefers sensitive information such as passwords, names, phone numbers, etc. This player represents banks, and insurance companies. The infrastructure player prefers sensitive network information, such as internet addresses, network architecture, Transmission Control Protocol (TCP)/Internet Protocol (IP) ports and services, etc. This player represents telecommunication companies or Internet Service Provider (ISP)+. The node player prefers sensitive computer equipment information, such as hardware addresses, computer configurations, file names, encryption keys, etc. This player represents the end user such as businesses and home users. This model represents the logging and network status sensing normally used in enterprise networks (e.g. security audit logs, intrusion detection system, etc.). This is done by allowing the defender to carefully observing the payoffs the defender receives. The defenders have three possible actions: social engineering defense, infrastructure defense, and node defense. Attackers have three possible actions: social engineering attack, infrastructure attack, and node attack. Payoffs are represented in US dollars for each type of information. This work uses players to represent each component in a complex network, such as the internet. The attacks are abstract and do not get into detail.

Tait [348] also presented an information warfare model. In his information warfare model there are four main elements: players, payoffs, information, and strategies. A set of players consists of an attacker and a defender. Each player a limited number of resources to use (funds, strategies, manpower, etc.). Each player

is assumed to be rational and will seek to maximize his or her expected payoff. The attacker's strategies include: attack system integrity, attack system confidentiality, or attack systems availability. The defender's strategies include: defend system integrity, defend system confidentiality, or defend systems availability. Each player may also choose to not implement a strategy. In this model the strategies are symmetrical across players, but the actions selected to carry out each strategy may be different. This work is based on the CIA model: confidentiality, integrity, and availability, which is used to determine network and data security. An important piece of this work is the constraint of limit resources available to deploy an agent.

Vejandla, *et al.* [358] proposed a method for generating gaming strategies for the Attacker-Defender game using evolutionary approach. The model takes into account objectives like cost, time, reward and performance. They use a memory-based Multi-objective evolutionary algorithm (MOEA) to generate the series of actions with the highest payoffs. The MOEA performed better than the existing approaches used to currently solve anticipation games.

Monderer and Tennenholtz [243] introduced the idea of Distributed Games. In this model, each player controls a number of agents, although the agents do not appear in the formal definition of the distributed game. Each agent participates in an asynchronous parallel multi-agent game and there is one agent for each location (node). Agents communicate by broadcasting messages. The Prisoner's Dilemma is played at all locations by each agent based on the messages received from other agents. The overall goal of this research is to show the cooperative nature of the agents.

Kivimaa, *et al.* [196] have presented an expert system for modeling graded security. Graded security is intended to determine a reasonable set of security based on a set of security requirement levels. The goals of security are confidentiality,

integrity, availability, and satisfying mission criticality. Four levels are used for each goal: 0, 1, 2, 3. This represents the required security; level 0, represents the absence of requirements. Nine security measures are used to achieve the security goals: user training, antivirus software, segmentation, redundancy, backup, firewall, access control, intrusion detection and encryption.

Several applications over the last few years have used multi-agent systems to allocate limited resources in order to protect mission critical infrastructures [20, 30, 184, 200, 281]. These multi-agent systems have been used for the Los Angeles International Airport, U.S. Federal Air Marshals Service, U.S. Coast Guard, U.S. Transportation Security Agency, and Urban Security in Transit Systems. Multi-agent systems are also being applied to protecting forests [187] and police patrols for crime suppression [268].

2.1.7 Deception in Game Theory.

Deception in game theory has been mostly studied in turn-based, or dynamic games, where a player choose an action then reports the action or outcome to the other player. This type of game is called signaling games, after the "signal" sent between players. The signal is subject to deception, since the player can be truthful, deceptive, or choose not to send a signal.

Carroll and Grosu [67] study network defense using deceptive signaling games. In their research, the defender can disguise a normal computer as a honeypot, a honeypot as a normal computer, or use no disguising techniques. The attacker has the ability to test the system type and the defender sends the appropriate signal, deceptive or truthful. The authors showed that deception is an equilibrium strategy for the defender, either by disguising all honeypots as normal computers or all normal computers as honeypots, provides an increase in utility for the defender over using only truthful signals.

Multi-turn attacker-defender games are used by Zhuang to study deception [376]. In the game, a defender type is randomly selected from a set of possible defender types and at each turn of the game the defender selects a strategy and "signals" the attacker of the selected strategy. The defender may be either truthful or deceptive. The attacker then uses the signal to update his belief of the defender's true type and selects an attack strategy. After each turn the payoffs are used to update the belief state until the game ends. The authors state, given their game, deception can be a beneficial strategy for the defender.

Hespanha, *et al.* [82] modeled an attacker-defender game where the defender has three units available to defend two locations. In the game the defender signals the locations of the units either by sending a truthful or deceptive signal or not camouflaging the units revealed to the attacker. The authors also discuss the possibility of a malfunction of the either the attacker's sensors or the defender camouflage, which may mean the signal seen may not be correct. The authors conclude that the use of deception can render the information collected from sensors and other methods to be useless to the attacker.

Deception has also been studied in repeated games. In this type of game the players both choose an action and make their moves simultaneously. Depending on the game, the players may receive information about how the environment stat changed between selecting moves. Pursuer-evader games are commonly modeled with this type of repeated game. Yavin [370] studies pursuer-evader deception, where both players choose a strategy based on the bearing of the other player and the distance between them, by corrupting the evader's bearing signal to the pursuer. The author's goal is to determine the optimal (or near-optimal) pursuit strategies for a pursuer when faced with deceptive or incomplete information.

2.1.8 *Bounded Rationality.*

Bounded rationality is where a players rationality is limited in the decision making process by the information the player has, cognitive limitations of their minds, and time available to make the decision [330]. H.A. Simon originally proposed the concept of bounded rationality as an improvement to the model of human decision making [329]. Bounded rationality helps to explain why the most rational decision is not always the decision chosen by the player in game theory or decision theory.

Bounded rationality does not mean irrationality, since players want to make rational decisions, but cannot always do so [188]. Players are often very complex, but in order to be fully rational they need unlimited cognitive capabilities [321]. The cognitive capabilities of players are limited and therefore cannot conform to full rationality. Players will use the cognitive resources they have, with the information available, and often within time constraints to reach a decision that is as rational as possible. Bounded rationality allows the player to make a decision based on their perceived state of the game or environment, leading to multiple players having different perceptions of the game or interaction.

2.2 Decision Theory

This section provides an overview of decision theory, as decision theory is one method to theorize about decision-making. In any given situation, there are actions which an agent can choose between and make a choice in a non-random way. The choose between actions are goal-directed activities [157]. Given a set of actions, decision theory is concerned with goal-directed behavior to reach a desired outcome.

2.2.1 *Overview.*

Decision theory is a formal mathematical theory about how decision-makers make rational decisions. It is also known as normative decision theory [280, 301], Bayesian decision theory [157], rational choice theory [319], and statistical decision theory [47].

Decision theory predates the development of game theory. Decision theory can be divided into three parts: normative, descriptive, and prescriptive [26, 145].

- Normative Decision Theory [280, 301] - studies the ideal agent and the decisions this perfectly rational agent would make, often referred to as the study of how decisions should be made.

- Descriptive Decision Theory [340] - studies the non-ideal agent, such as humans, and how they make decisions, often referred to the study of how decisions are made in reality.

- Prescriptive Decision Theory [161] - studies how non-ideal agents, given their imperfections, can improve the decisions they make.

2.2.2 Basis For Theory.

Normative decision procedures are defined by beginning with some axioms of rational decision making behavior, then use the axioms to derive a characterization of rational decision making [299]. Von Neumann and Morgenstern present the first axiomatization for decision theory. They give four axioms which an agents preferences must follow in normative decision theory [259]:

- Completeness - Each agent must have a preference for each pair of outcomes or be indifferent between the two.

- Transitivity - If A is preferred to B and B is preferred to C, then A is always preferred to C.

- Independence - Preferences hold independent of any other possible outcomes. If A is preferred to B, then A is always preferred to ApB (read as A with probability p else B), which is preferred to B.

- Continuity - If A is preferred to B, then when given the choice between CpA (read as C with probability p else A) or CpB, then CpA is always preferred to CpB.

These four axioms lead to the conclusion that "a rational decision maker will act according to their degree of belief which conforms to probability calculus"[100]. For example, a decision theorist may describe a rational decision maker as having the following characteristics [299]:

1) A rational decision-maker's behavior is guided by their degree of belief in the occurrence of an event. For example, if a sports fan is offered the choice to win $500 if their team wins this week or $500 if the team wins next week, but can only select one option. A rational sports fan would make their decision based on the belief of the team's best chance to win, given the opponent, weather conditions, etc.

2) When given the option between being guaranteed breaking even in a game or the possibility of breaking even with a chance of losing, a rational decision maker will always select being guaranteed breaking even.

These five characteristics lead to a rational decision maker that acts with the following behaviors [157, 356]:

1) has a set of beliefs represented by a probability distribution (P) over all possible outcomes

2) assigns an utility (U) to every possible outcome using an utility function that results in preferred outcomes receiving higher utilities

3) selects a strategy or action (A) by determining the expected utilities (EU) and selecting the highest EU, thus maximizing expected utilities

2.2.3 Bayesian Decision Theory.

Bayesian decision theory, or Bayesianism, is based on subjective utilities and subjective probabilities. It can be described by the following four principles, where the first three refer to the agent with probabilistic beliefs and the fourth refers to the agent as the decision-maker [157].

- Coherent set of probabilistic beliefs - where the agents beliefs comply with the mathematical laws of probability.

- Complete set of probabilistic beliefs - where each outcome is assigned a subjective probability

- Beliefs are updated according to the agent's conditional probabilities - that is beliefs are updated according to Bayes' Rule: p(A—B) = p(A&B)/p(B).

- The outcome with the highest expected utility is always chosen.

In descriptive Bayesianism, decisions made by decision-makers satisfy the previous four principles. With normative Bayesianism, rationality is key, decisions made by rational decision-makers satisfy the previous four principles. Subjective Bayesianism does not present a particular relationship between the subjective utilities used or the objective frequencies, such as a coin flip.

Bayesianism is not as popular in practical decision situations. This is because of the subjective probabilities and utilities are hard to test. Objective probabilities and utilities lead to predications that can be validated through testing.

Bayesian games have been proposed by Harsanyi [160]. Harsanyi claimed a game of incomplete information can be captured by subjective probability distributions over the outcomes without loss of generality. This is done by introducing a set of types and each player's belief about the types. Bayesian games, based on Harsanyi's claim,

allow all players to share a common set of the possibilities of the game structure [313]. This claim is often controversial as real interactive situations may lead to possibilities that are not shared by all players [138].

2.2.4   Weakness of Decision Theory vs. Game Theory.

The "outguessing problem" is based on a player's worry of what the other player may do. There is nothing in the decision theory axioms that would encapsulate the outguessing problem. This would imply worrying is not rational under decision theory. A normative decision maker has a probability distribution over all possible outcomes, and a probability distribution of the adversary's behavior given the possible actions. Unlike game theory, decision theory does not have a way to predict a column's action based on the column's understanding of the row's strategy [356]. In this case decision theory does not require columns to select actions according to rational behavior; it only requires that a row has the ability to assess the intent of column during the game.

It is possible for the decision-maker to use the two different approaches during game play. For example, decision-makers may use a risk adverse approach where the agent prefers actions that lead to less risk in the expected outcome values, where the agent could also use a risky approach where the agent prefers actions that lead to more risk in the expected outcome values, but also higher expected outcome values. This is especially realistic in retirement planning, a person normally starts out risky, choosing investments with high risk and volatility, but with high reward. As the person ages and approaches retirement, they start accepting less and less risk. Under decision theory, modeling a decision-maker with two different approaches is difficult.

None of the decision theory axioms determine how the probability distributions are derived. It is possible to model a decision-maker that uses multiple approaches the decision-maker could use game theory to gain insight into the opponent's behavior by

using game theory to derive the probability distributions and explain the opponent's behavior.

Decision theory lacks the ability to quantify the risk of being outguessed during a game. During a game, belief weights are associated with the column that results in the least desirable outcome, since the belief weights corresponding to the most dangerous column may be less than one [356]. This may lead to actions being discarded that would help mitigate the worst-case condition. This idea stems from the fact that in the real world, players may not be able to reach their preferred or most desired outcome because their opponent has an unknown action or information, but at the same time they could guard against reaching their least preferred outcome by taking risk into account.

None of the four axioms in normative decision theory determine how to derive the probability distribution corresponding to the beliefs of the opponent player. This means that playing a game according to the player's beliefs does not make any decision irrational, but the decision may be ineffective if the beliefs used as the basis for the probability distribution do not correspond to reality, leading to a delusion for the player.

Within decision theory it is implied that a probability distribution can be determined by the decision-maker [356]. This is without regard for the number of actions, strategies, players, or opponents. It is also implied the probability distribution of the situation it models. In real world events it may be hard to create realistic probability distributions, do the possibility of asymmetric information.

2.2.5 Summary.

This section discussed decision theory as a method to theorize about decision-making. It requires a probability distribution to be derived by the decision-maker. The probability distribution must be consistent according to the axioms of decision

theory, but decision theory does not specify how to update the probabilities. This leads to Bayes' rule being used in Bayesian decision theory to update the probabilities. This allows decision theory based on a set of actions, given the axioms, to reach a desired outcome using goal-directed behavior.

2.3 Game Theory vs. Decision Theory

Game theory is a bag of analytical tools designed to help understand the phenomena that observed when decision-makers interact [270]. Decision theory is a formal mathematical theory about how decision-makers make rational decisions as they interact with their environment. What is the difference between game theory and decision theory? There is a division between decision theory where the outcome depends on the players decisions and the impersonal universe, while game theory depends on the decisions made by interacting with other players.

2.3.1 Overview.

Throughout the literature on decision theory and game theory there are slightly different views on the division between the two theories. As shown in Figure 2.9a - 2.9d there are four possible high level views of the division between decision theory and game theory. While it is possible decision theory and game theory are completely independent of each other (as shown in Figure 2.9a, this is not considered in the literature. The application of the theories also does not support this.

It is possible for game theory to be part of decision theory, as shown in Figure 2.9b. Given the definitions of decision theory and game theory from the literature, this does not seem likely. Most definitions define decision theory as more specific theory than game theory. For example, the definitions presented in this appendix define game theory as a bag of tools and decision theory as a mathematical theory or tool.

Some literature considers some parts of decision theory and game theory to be distinct, while there is some overlap between the two, as shown in Figure 2.9c. Often the opponent of the player in question is different between the two (with decision theory focusing on a player against nature and game theory focusing on the interaction of evenly matched players). There are some characteristics shared by both theories. For example, decision theory and game theory make use of rationality and preference ordering, as well as probability theory.

(a) Independent Theories

(b) Game Theory part of Decision Theory

(c) Related Theories

(d) Decision Theory part of Game Theory
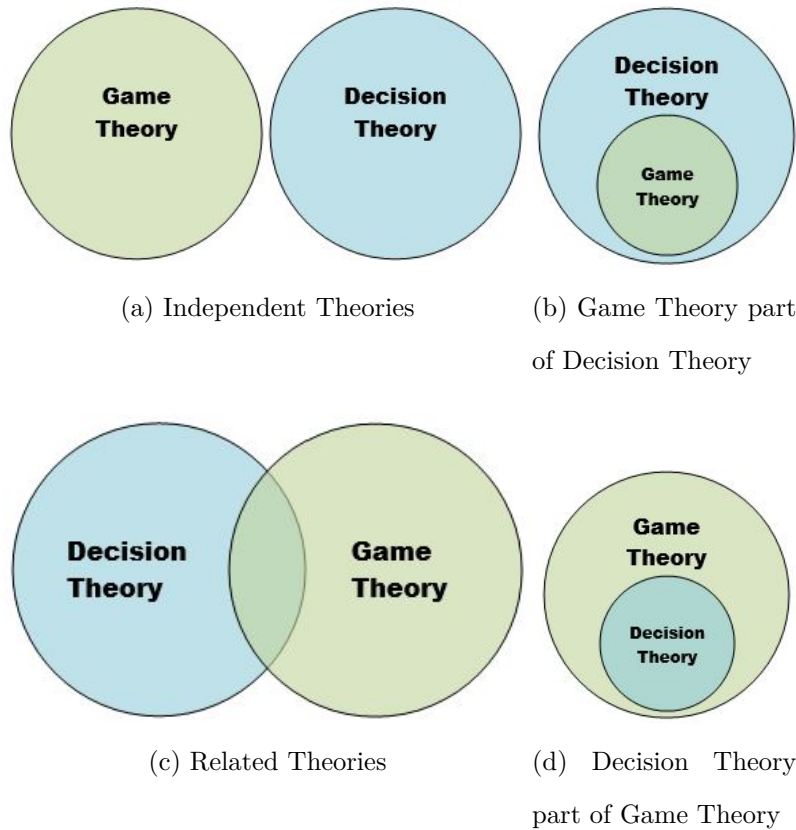
Figure 2.9: Possible Division of Game Theory and Decision Theory.

It is also possible for decision theory to be part of game theory, as shown in Figure 2.9d. Given the definitions of decision theory and game theory from the literature,

this is possible. This is due to the fact that decision theory is defined as a more specific theory than game theory. Using the definitions presented in this appendix, it is easy to see that if game theory is a bag of analytical tools and decision.

Often the division appears to be arbitrary. The arbitrary division appears when the players in the game are borderline players, such as animals or small children. Other players could only be interacting minimally, such as workers in a building and facilities management. The rest of this appendix discusses the differences between decision theory and game theory.

2.3.2 A Short Story.

One of the easiest ways to see the differences between game theory and decision theory is through an example. While this story is a bit contrived, it does show differences in decision-making. The following short story, taken from Scientific American, illustrates the difference between game theory and decision theory [351]:

> A hat seller, on waking from a nap under a tree, found that a group of monkeys had taken all his hats to the top of the tree. In exasperation he took off his own hat and flung it to the ground. The monkeys, known for their imitative urge, hurled down the hats, which the hat seller promptly collected.
>
> Half a century later his grandson, also a hat seller, set down his wares under the same tree for a nap. On waking, he was dismayed to discover that monkeys had taken all his hats to the treetop. Then he remembered his grandfather's story, so he threw his own hat to the ground. But, mysteriously, none of the monkeys threw any hats, and only one monkey came down. It took the hat on the ground firmly in hand, walked up to the hat seller, gave him a slap and said, "You think only you have a grandfather?"

In the story, the grandfather uses decision theory to reach the decision that since monkeys will imitate actions, then to get his hats back he just has to throw his hat on the ground. Here decision theory would apply since the monkeys can be considered to be acting naturally given their nature. The grandfather makes his decision without assuming or considering the monkeys are rational or think like himself.

The grandson, on the other hand, uses decision theory to reach the decision to throw his hat on the ground, recalling his grandfather's story. While using decision theory, the grandson never considered the monkeys as strategic decision-makers. If he would have used game theory, he would have reasoned that if he learned the hat trick from his grandfather, then the monkeys would have learned the hat trick from their grandfathers.

2.3.3 Questions From the Theories.

Decision theory deals with single player games, or games where a player is against nature, with the focus on preferences and the formation of beliefs [220]. The implicit assumption in decision theory is nature is not cheating, interfering with, or assisting the player; nature continues to function without regard to what the player wants wishes to accomplish.

The main focus of decision theory is on one question:

- How do individuals make decisions?

Game theory deals with multi-player games, often involving groups of people, where players are in cooperation or competition with competing strategies. It explicitly assumes the other players are rational and may be cheating, interfering with, or assisting the player or other players. In game theory it is necessary to interact with other rational and intelligent players in order to resolve the conflict.

The focus of game theory can be summarized with two questions about the interaction of the decision-makers [364]:

- How do individuals behave in strategic situations?

- How should these individuals behave?

2.3.4   The Differences Between Game Theory and Decision Theory.

The first difference is how the decision is made to select an action with two actions that have equal expected utilities[356]. Decision theory sees both actions as desirable, so it would be rational for either action or a combination to be chosen. This follows directly from the axioms of decision theory. In game theory, the other player(s) is considered and with this in mind, even with two actions of equal expected utilities, the player may prefer one or the other. For example, in a conflict with an enemy, choosing the one action out of the two, which leads to a lower expected value for the opponent would be preferred. Otherwise, if the player were playing family member, they would prefer the action that leads to the higher expected value for the opponent.

The second difference is how to handle when a player is wrong about their opponent's intent. Generally in decision theory, if there are two different views, then these views are combined in the probability distribution. This leads to a player's best guess always being used, which is the only rational behavior [314]. In game theory, two different views can be considered during game play and become part of the rational reasoning expected with game theory. This allows the game theory based solution to maximize the expected value while minimizing the chance they are wrong about their opponent's intent.

2.3.5   Where the Two Meet.

The foundations of decision theory do not guide how the probability distribution are derived. In order to model a decision-maker that uses more complex interactions, the decision-maker could use game theory to gain insight into the opponent's behavior by using game theory to derive the probability distributions and explain the

opponent's behavior. In this environment, game theory adds value to the decision analysis through its suggestion on how the decision maker should form its beliefs about the behavior of the environment, i.e., about the actions of players whose behavior is modeled as uncertain [50].

2.3.6 Summary.

While game theory and decision theory are similar on the surface, after a closer look they are different. Game theory focuses on multi-player games often involving groups of people where players are in cooperation or competition with competing strategies. While decision theory focuses on single player games or games where a player is against nature with the focus on preferences and the formation of beliefs. This chapter discussed the differences between game theory and decision theory, as well as the similarities between the two theories.

III. Hypergame Preliminaries

This chapter describes the background and related work for understanding hypergame theory. Section 3.1 provides background on hypergame theory focusing on its ability to model complex conflicts with unbalanced information. Section 3.2 covers application of hypergames where competitive nature and proprietary information often lead to missing information and a desire to introduce misperceptions (such as in military conflicts, sports, resource allocation, business, and cyber). Related work in hypergame theory is discussed in Section 3.3.

3.1 Hypergame Theory

Game theory, decision theory, and hypergames can be used to model conflicts as games. When very little is known about the opponents, game theory is used for adversarial reasoning. Decision theory is a better choice if the opponents are well known. If one or more of the opponents are playing different games because they are not fully aware of the nature of the game, hypergames can be used to reason about subgames that are shared between opponents. Decision theory is not discussed in the following section because in most cyber operations the opponents are not well known. This may be due to the trouble of attribution, government operations that are classified, or the global nature of the Internet.

"A conflict is a situation in which there is a condition of opposition [116], and parties with opposing goals affect one another [106]." The study of how decision makers interact during a conflict is known as game theory. An overview of game theory and its applications are given in Chapter 2. Game theory analysis often falls short when one player has an advantage over the other in a conflict. When one or more players lack a full understanding, have a misunderstanding, or incorrect view of the nature of the conflict, hypergame theory can be used to model the conflict.

Hypergames extend game theory by allowing for an unbalanced game model that contains different view of the game representing the differences in each player's information or beliefs. The unbalanced game model allows for a different game model for each player's view, while having overlap where there is common knowledge. The solution to the hypergame model is dependent on the player's perception of the game model, including how the player views the game and how the player believes the opponent is viewing the game. Because of the multiple game models, each model has to be analyzed in order to determine the outcome to the hypergame. This allows hypergames to more accurately provide solutions for complex real world conflicts than those modeled by game theory and excel where perception or information differences exists between players.

3.1.1 Decision Making and Learning.

There is a feedback loop between experience and views in a hypergame [310]. This feedback loop is taken from [310], extended with hypergame concepts, and is shown is shown in Figure 3.1. Every player in a hypergame starts with a set of perceptions. A player's perceptions consist of beliefs, preferences, knowledge, and subgames. Beliefs are based on past experiences and the player's environment. Preferences represents the player's preferred ordering of actions and preferred results of the playing the game and it is unique to each player. Knowledge of the game (which may be faulty or incomplete) is important to forming player perceptions. A player uses orientation and observation, along with the player's perceptions in order to make a decision. The player orients their view of the environment and observes the experiences of the game.

Each player has a set of experiences. The experiences can consist of deception, subversion, payoffs, and denial. Players may be aware of certain aspects of their experiences, such as payoffs, and may be unaware of other aspects such as deception or subversion. Experiences provide feedback and elements of surprise in order to allow

54
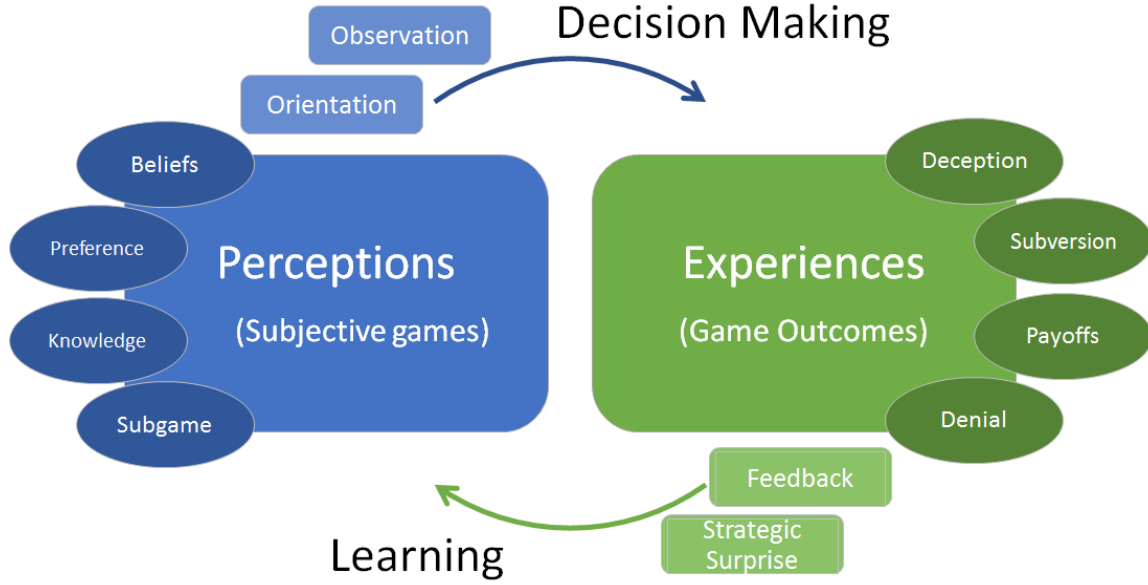
Figure 3.1: Relationship Between Decision Making and Learning.

a player to learn and update/change their perceptions throughout the course of the game.

This process when used in real life conflicts requiring decision making, often results in the process retaining temporal aspects. Often a time dependence develops and becomes an integral part of the decision making process.

3.1.2 Hypergame Basic Concepts.

Hypergames, first discussed by Bennett [34], are used to model the games where one or more players are playing different games [43]. Hypergame theory decomposes a single situation into multiple games. By reasoning about multiple games, the outcome to the single problem can be improved. Each player in a game has their own perspective of how the other players view the game with regards to the possible actions, and player preferences. In a hypergame each player may [106]:

- have a false or misled understanding of the preferences of the other players

- have incorrect or incomplete comprehension of the actions available to the other players

- not have awareness of all the players in a game

- have any combination of the above; faulty, incorrect, incomplete, or misled interpretations

A player's choice of actions reflects the player's understanding of the game outcomes; the player chooses actions based on the way they perceive reality, which may not be the true state of reality. Figure 3.2 shows a basic two player hypergame between "row" and "column", where C_i and R_i are different actions each player could take.

Figure 3.2: Example of a hypergame.

Hypergame analysis is conducted by first examining Row's belief about Column's reasoning, and then by examining Row's available actions [353] [354]. In Figure 3.2, the game on the left shows how Row thinks Column will reason about the game. Based on this Column will play C_2 while Row plays R_2, the Nash equilibrium concept from game theory. This allows the experience and intuition of the decision maker to be

incorporated into hypergames. For example, this could apply to planning variables, such as a novel course of action for Row or Column's lack of time to plan, or to situational variables, such as the hidden location of Row's resource [106].

Hypergames allow for domain knowledge incorporation, therefore it does not require the game theory equilibrium condition [106]. Furthermore, the standard rationality arguments from game theory are replaced by knowledge of how the opponent will reason [106]. It is also valid to assume unequal availability of information in hypergames, when many players in games have imperfect information.

Wang *et al.* [361] proposed different levels for developing mathematical hypergame models based on perceptions of the players. The lowest level (level 0) is a basic game with no misperceptions among the players. In a first level hypergame, players have different views of the game but are not aware of the other players' games. In a second level hypergame, at least one player is aware there are different games being played and that misperceptions exist. A third level hypergame is possible and is when at last one player is aware that at least one other player is aware different games are being played. A nth level hypergame can be described, but the authors state this does not add to the hypergame model, instead it adds complication and excess information not needed for the hypergame analysis. This not only allows the perceptions of the players to be incorporated into the hypergame model, but varying degrees of perceptions in order to reach a more complete game model.

3.1.3 First Level Hypergame.

A game G is defined by a set of preference vectors, V_n , for all game players; where n is the number of players, and V_i is the preferences vector for player i.

$$G = \{V_1, V_2, \ldots, V_n\}$$

In game of complete information, all players know the other player's preference vectors, therefore all players are playing the exact same game. In hypergames, one

or more players may have incomplete information, that leads players to form slightly different versions of the same game or completely different games altogether. A game formed by player q includes any and all lack of information about the conflict, which is denoted by:

$$G_q = \{V_{1q}, V_{2q}, \ldots, V_{nq}\}$$

Where V_{iq} represents the preference vector of player I as understand(perceived) by player q.

A first level hypergame H is a set of games as understood from each player:

$$H = \{G_1, G_2, \ldots, G_n\}$$

Table 3.1 shows a hypergame in matrix form.

Table 3.1: Matrix form of a hypergame.

Player Perceived	Game perceived by player			
1	V_{11}	V_{12}	$\ldots$	V_{1n}
2	V_{21}	V_{22}	$\ldots$	V_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	V_{n1}	V_{n2}	$\ldots$	V_{nn}
	G_1	G_2	$\ldots$	G_n

Since players may have different misperceptions, each player may make a different decision which will result in a different outcome to the conflict. A mapping function can be used to relate the outcomes between the player's individual games. Bennett [36] gives an algebraic description of this problem, while an application is presented in Bennett et al. [42].

Game analysis is performed by treating each player's game separately. This means player q's game is analyzed from q's understanding about the conflict. The decisions made and the strategies chosen by q depend on q's interpretation of the conflict, therefore a given player may not perceive all outcomes of a game. The player cannot unilaterally change from an perceived outcome, so for the purpose of stability analysis the outcome is stable for that player [106]. Therefore an unknown outcome to a player can be stable in the hypergame analysis. When a game contains an unknown outcome it is known as strategic surprise.

For player q's game, an outcome is stable if the outcome is stable in each of q's preference vectors. This means the equilibriums of q's game are only the outcomes q believes would lead to a resolution of the conflict. Hypergame equilibriums depend on each player's perception of the stability of the outcomes. When determining equilibriums of hypergames, the equilibriums of each player's game are not needed, but these individual equilibriums can be useful to demonstrate what each player believes will happen.

3.1.4 Second Level Hypergame.

A second level hypergame is a hypergame where at least one player is aware a hypergame is being played. This situation can happen if at least one player perceives another player's misperception [106]. Player q's hypergames is defined as the (hyper) game perceived by player q. This hypergame is denoted as:

$$H_q = \{G_{1q}, G_{2q}, \ldots, G_{nq}\}$$

Where G_{iq} is the game of the ith player as it is perceived by player q. It is not necessary for player q to be one of the players who are aware a hypergame is being played. If set Hq is missing a player's game, it is because player q does not perceive the game.

59

A second level hypergame is a set of hypergames perceived by each player, denoted as:

$$H^2 = \{H_1, H_2, \ldots, H_n\}$$

Table 3.2 shows a second level hypergame in matrix form, where the hypergame for player p is the pth column. Each element of the matrix is a game made up of a preference vector for each player.

Table 3.2: Matrix form of a second level hypergame.

Player Perceived	Game perceived by player			
1	G_{11}	G_{12}	$\ldots$	G_{1n}
2	G_{21}	G_{22}	$\ldots$	G_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	G_{n1}	G_{n2}	$\ldots$	G_{nn}
	H_1	H_2	$\ldots$	H_n

Similar to a first level hypergame analysis, game analysis of second level hypergames is performed by treating each player's game separately. This allows stability information to be determined for every preference vector in a conflict. This information can further be used to determine each game's equilibrium.

The preference vectors of each player's game provides the stability information that determines the equilibriums of the second level hypergame. "Just as the equilibriums of a game within a hypergame are not needed to determine the equilibriums of that hypergame, so the equilibriums of a hypergame within a higher level hypergame are not needed to determine the equilibriums of that higher level hypergame." [106]

3.1.5 *Hypergame Normal Form.*

Russell Vane [356] incorporates a player's beliefs with the opponent's possible actions in the hypergame model. Vane refers to this as Hypergame Normal Form (HNF) this is based largely on the strategic from used in standard game theory analysis, as shown in Figure 3.3. Like the strategic form, HNF contains a grid with row and column strategies labeled and utility values defined for each cell where strategies intersect.

			Belief-contexts				C_Σ	CMSs S_1	S_2	S_3	...	S_n
			P_{K-1}	...			C_{K-1}	C_{k1}	C_{k2}	C_{k3}	...	C_{kn}
				...			...	...	...	...	...	...
				...	P_1		C1	C_{11}	C_{12}	C_{13}	...	C_{1n}
Hyperstrategies				...		$P_0 = 1 - \sum_{i=1}^{K-1} P_k$	C0	C_{01}	C_{02}	C_{03}	...	C_{0n}
PS	WS	MO	R_{K-1}	...	R_1	$R_0 =$ full game		Column 1	Column 2	Column 3	...	Column n
PS_1	WS_1	MO_1	r_{k1}	...	r_{11}	r_{01}	Row 1	u_{11}	u_{12}	u_{13}	...	u_{1n}
PS_2	WS_2	MO_2	r_{k2}	...	r_{12}	r_{02}	Row 2	u_{21}	u_{22}	u_{23}	...	u_{2n}
...	...	...	...	...	...	...	...	...	...	...	...	...
PS_m	WS_m	MO_m	r_{km}	...	r_{1m}	r_{0m}	Row m	u_{m1}	u_{m2}	u_{m3}	...	u_{mn}
EU_{PS}	EU_{WS}	EU_{MO}	$EU_{R_{K-1}}$	...	EU_{R_1}	EU_{R_0}	EU	← Expected Utility	Full Game			
G_{PS}	G_{WS}	G_{MO}	$G_{R_{K-1}}$	...	G_{R_1}	G_{R_0}	G	← Worst Case				
HEU_{PS}	HEU_{WS}	HEU_{MO}	$HEU_{R_{K-1}}$	...	HEU_{R_1}	HEU_{R_0}	HEU	← Hypergame Expected Utility				

RMSs

Figure 3.3: Hypergame Normal Form (HNF) [356].

The HNF grid has additional sections which contain hypergame situational information. This hypergame situational information is represented by Row Mixed Strategy (RMS) and Column Mixed Strategy (CMS). A RMS is a hyperstrategy based on what the row player believes in the game being played by the column player. Hyperstrategies are strategies that do not apply to the full game, except for the hyperstrategy R_0 which represents the Nash Equilibrium for the full game. A

61

CMS represents row's beliefs about the percentages column will use when selecting a strategy. The Nash Equilibrium for column for the full game is represented by C_0. A CMS cell containing a zero (0) indicates column is playing a subgame with strategies that are either unknown or disregarded. Another section contains belief contexts represented as the percentage that row believes the adjacent CMS will be selected and played by column. Belief contexts will sum to one since they represent all possible strategies from row's view, any remainder corresponds to the Nash Equilibrium belief context.

The HNF is completed by first calculating the Nash Equilibrium for the full game by determining the utility values, which is the input for R_0 and C_0. The CMSs can be calculated manually by knowing the player's preference for the strategies or by finding the Nash Equilibrium of a subgame for the column player. A weight is assigned to each CMS as a belief context value, which is used to determine row's belief of column using that strategy. These weights are then used to calculate the amount that affects row's expected utility, C_Σ. Hyperstrategies can be placed in the RMS section for the row player. Expected utility values are calculated for the full game CMS, C_0, and C_Σ in order to determine the effectiveness of which RMS hyperstrategy row should select.

The effectiveness of RMS hyperstrategies is measured using three levels: ineffective, partially effective, and fully effective. These levels are shown in Figure 3.4 along with the Nash Equilibrium Mixed Strategy (NEMS). Ineffective strategies do not increase utility, leading to a best case outcome of the Nash Equilibrium. This means there is no reason for the player to choose an ineffective strategy. Partially effective strategies lead to greater expected utility at C_Σ than at R_0, while providing a lower expected utility at C_0. A partially effective RMS may provide a good outcome based on row's information, but a good outcome is not guaranteed. Fully effective

62

strategies lead to in the worst case the same expected utility R_0 results in for C_0, but with greater utility than C_Σ. If row's beliefs about the game are correct, then a RMS strategy that is fully effective is a reasonable strategy for row to play. These relationships are summarized in Figure 3.5. While a fully effective strategy is a reasonable strategy for row, it does not guarantee this is the best strategy in all cases. The utility values are expected and not actual utility values. In order to mitigate the risk when using expected utility values, the worst case scenario can be used to select the strategy.
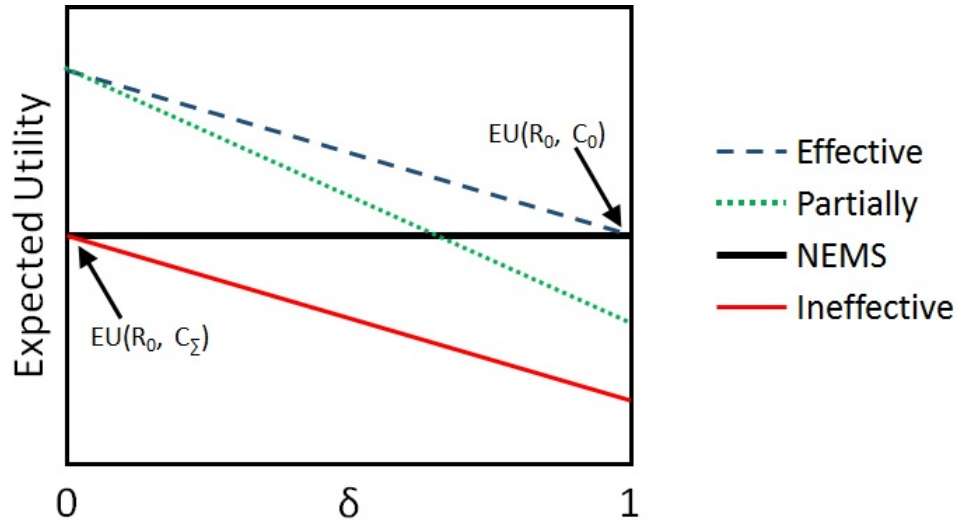
Figure 3.4: Effectiveness of Hyperstrategies in HNF [356].

Vane uses quantified outguessing in order to measure a players adversity to risk. This measure quantifies the player's fear of obtaining the worst outcome or lowest utility from the game by being out maneuvered. There are three types of hyperstrategies for analyzing quantified outguessing: Modeling Opponent (MO), Pick Subgame (PS), and Weighted Subgame (WS). In MO, row's strategy that provides the greatest utility given all of row's strategies and row's belief of how column is

63
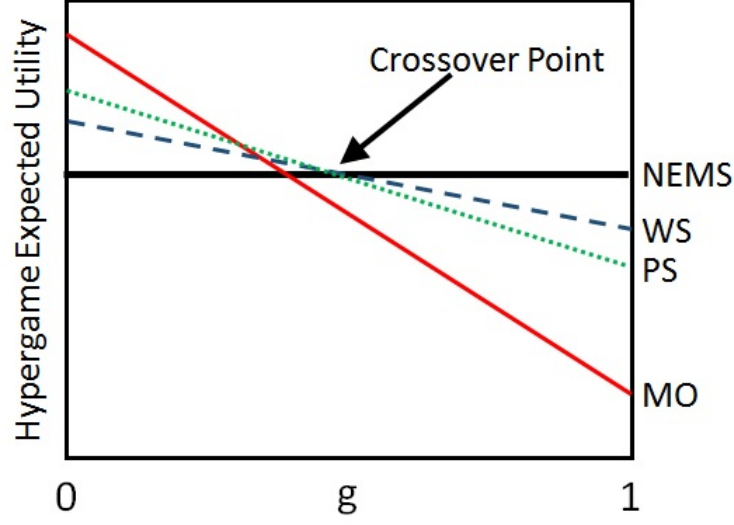
Figure 3.5: Relationship of Hyperstrategy Effectiveness [356].

viewing the game. In PS, row's strategy is the Nash Equilibrium given the same game considered in MO. The WS strategy takes the PS strategy values and multiplies by the belief context percentage for the given CMS then adds the R_0 multiplied by the belief context for each C_0. This produces a hybrid strategy between PS and the Nash Equilibrium Mixed Strategy of the full game. Each of the hyperstrategies are then analyzed against the full game to obtain the worst case utility (G), which is the expected utility value if column selects the best counter strategy. The Hypergame Expected Utility (HEU) can be calculated using the Expected Utility (EU) and G, along with the percentage, g, that represents row's fears of being outguessed based on the hyperstrategy (hs) as shown in the following equation:

$$HEU(hs) = EU(hs) - (EU(hs) - G(hs)) * g$$

A hyperstrategy's ability to provide a better utility at the Nash Equilibrium Mixed Strategy decreases as row's fear of being outguessed increases, as shown in Figure 3.5. MO is the best solution when the fear of being outguessed is low. As

64

the fear of being outguessed increases PS dominates until the Crossover Point. At the Crossover Point the Nash Equilibrium Mixed Strategy of the full game becomes dominant. WS does not provide a suitable choice as it is always dominated. It is possible to select hyperstrategies with higher utility than standard game analysis, with better information about the intents of the opponent.
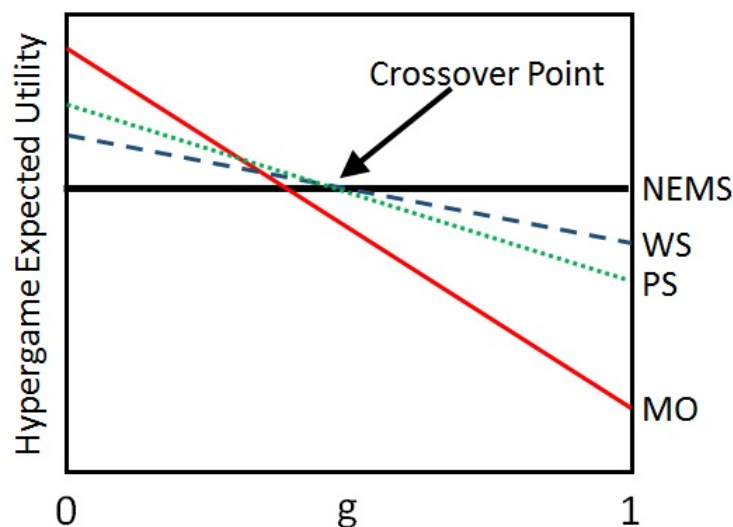
Figure 3.6: The Effect of g on the Hypergame Expected Utility (HEU) [356].

In hypergame theory subgames represent a smaller game that differs in a key way from the larger main hypergame. Often subgames differ by different combinations of actions. Analyzing the subgames allows the player to see how the outcomes change as the game model changes. This allows the modeling of false or misleading understandings, incorrect or incomplete comprehension, lack of awareness, and faulty interpretations of the game.

As shown in Figure 3.7, subgames can differ by the column player having a different set of actions, which could lead to different outcomes/payoffs. A full game is defined in Definition 7 and a subgame is defined in Definition 8.
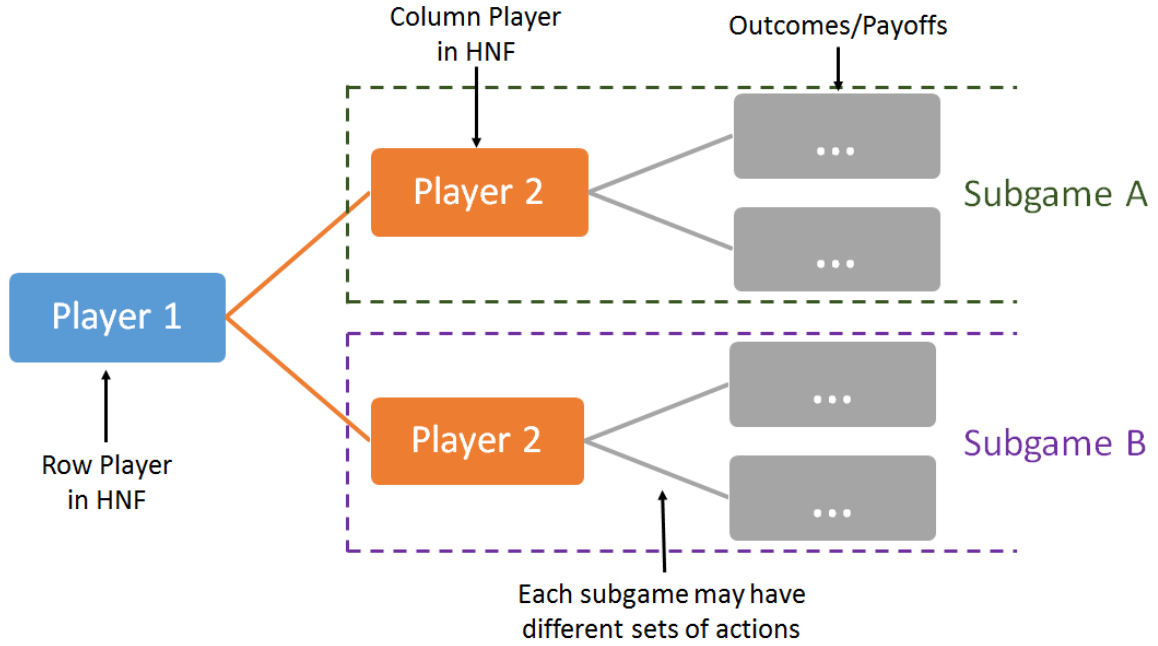
Figure 3.7: Hypergame Subgame Example.

Definition 7. ***Full Game Hypergame*** *- When the row player has m actions and the column player has n actions (called a m x n hypergame), it is called a full game hypergame.*

Definition 8. ***Subgame*** *- Given a full game hypergame, a subgame is a game defined with up to m rows and up to n columns (called a u x v subgame), where $u \subset m$ and $v \subset n$.*

The equation to determine the possible number of subgames is Equation 3.1 [356]:

$$\binom{m}{u} \cdot \binom{n}{v} = \left(\frac{m!}{u!(m-u)!} \right) \cdot \left(\frac{n!}{v!(n-v)!} \right) \qquad (3.1)$$

While a large m x n hypergame game could result in hundreds of subgames, limitations normally result in just a few subgames. Limitations include [356]:

- Time and effort

- The way human memory works

- Leaders distilling situation into a few courses of action

- Decision-makers condense plan selection problem into a handful of reasoning contexts

Subgames are important when players interpret the payoffs in the environment. The player's interpretation can be wrong. For example, in Figure 3.7, Player 2 may believe Player choose an action that leads to Subgame B, while reality is that Player 1 choose an action that lead to Subgame A. This means Player 2 continues playing the game according to Subgame B.

Vane expanded hypergame analysis by exploring the robustness of strategy plans [181]. This shows the ability of using hypergames for strategy selection. He also applied HNF to a real world example of a terrorist attack [180]. The research aims to pick the strategy to best protect first responders by applying the belief contexts to the types of attackers expected during an attack. HNF is also applied to the Fall of France in 1940 [32]. This application shows that information in the HNF model is not removed even if a strategy is discounted with a chance of zero. The strategy remains in the Nash Equilibrium Mixed Strategy and is not entirely removed from the model, showing the flexibility and robustness of the HNF model.

3.2 Applications of Hypergame Theory

Hypergame theory has been used to examine past military conflicts, which by their nature are conducted with missing information and misperceptions. Past

conflicts lend to analysis because the excitement and fog of war has cleared as well as the outcome has already been determined. Hypergame theory has also been applied to sports, resource allocation, and business, where competitive nature and proprietary information often lead to missing information and a desire to introduce misperceptions. Recently, it has been applied to cyber conflicts using attacker/defender models, where resource constraints and advantage are important.This section provides an overview of each type of application.

The applications of hypergames are separated into five distinct categories as shown in Figure 3.8: military conflict, sports, resource allocation, business, and cyber. These categories contain the majority of the hypergame application work. En overview of the numerous applications in hypergame theory is summarized in Table 3.3. Each is listed chronologically and denoted with the corresponding year and topic category.
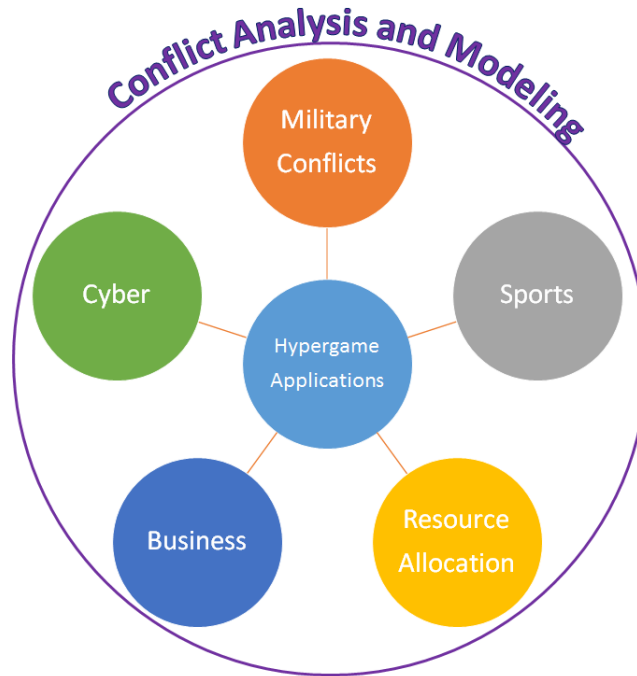
Figure 3.8: Hypergame Application Characterization.

3.2.1 Military Conflicts.

Bennett and Dando [39, 40] first applied hypergames to the first real world application during their analysis of the Fall of France during WWII. They show how misperceptions between the two countries can lead to unexpected outcomes using hypergames. The hypergame model they used is shown in Appendix A and discussed in detail.

Wright, *et al.* [328, 369] presented a more complex hypergame example in their analysis of the Nationalization of the Suez Canal in the 1950s. This hypergame shows how one player waiting to participate in the conflict can lead to strategies changing over time. While this is a temporal concept, the analysis is only made for one point in time during the conflict. The hypergame used for analysis is presented in Appendix A.

Said and Hartley [308] use hypergame theory to analyze the 1973 Middle East War. Their analysis shows that each player behaves in a rational manner within their own perceptual beliefs. The deailts of the 1973 Middle East War are let out, as the Fall of France and Nationalization of the Suez Canal are similar. The contribution of this work is the proposed methodology for applying hypergame theory to a crisis:

- Specify all conflict participants (individuals, groups, or organizations)

- Divide the conflict in phases, but only initially proceed with the first phase

- Model each player's perceptions of the conflict

  - List all the strategies a player perceives all players (including himself)having

  - Estimate the player's preferences for outcomes

  - Estimate the preferences of other players perceived by the current player

  - Use the resulting game to explore alternative modes of behavior a player may be expected to exhibit

69

- Repeat process for each player, resulting a set of possible games based on perception

- Map the strategies in the current player's game into the set of possible games in order to help correlate how each player perceives the actions of opponents.

Bennett and Dando [41] model an arms race between two nations as a hypergame. Their analysis forces he modeler to consider the perceptions, beliefs, and actions of ll parties involved, which they claim to lead to a more competent analysis. Appendix A has additional details on the arms race model.

Fraser, *et al.* [107] apply five conflict analysis models to a possible nuclear confrontation between the USA and USSR. The five conflict analysis models are normal form analysis from game theory, metagame analysis [175], and hypergame analysis [108] [106]. An overview of each of the models follows is in A. Their analysis determines that the hypergame analysis of conflicts is the best for modeling real-world conflicts.

Hipel, *et al.* [167] examine the Falkland/Malvinas conflict in 1982. The authors approaches the conflict from a different angle in their analysis of the conflict between Britain and Argentina. The hypergame analysis of the conflict is used to show how misperceptions dictated an outcome that was unexpected by all sides. This analysis uses three specific points in the conflict to construct three different hypergame models. The authors construct the hypergame model based on historical material, using a first-level hypergame, as discussed in A.

3.2.2 Sports.

In the literature there is only one example of hypergame being applied to sports. While this is far fewer than military or business conflicts, it is easy to see how the competitive nature of sports lends to hypergame modeling.

70

Bennett et al. model soccer hooliganism [42] which appears in U.K. soccer around the late 1970s. They use the hooligan fans and the authorities as the players. Empirical studies were used to build up possible games that may be played between the players. The hypergame analysis showed that there were three critical variables: (1) the fans interpretation of how the authorities prepared for possible conflict; (2) how the authorities interpret the "Play Hooligan" strategy by the fans; (3) the effect previous incidents have on perception for future conflicts. The result of the analysis is that tolerance should be used by the authorities. This reduces the over preparation and expectation everyone is a hooligan, and in time reduces the effect of previous incidents.

When the hypergame goes through a number of iterations, additional forces put pressure on players in the game. For example, previous incidents will place pressure on the authorities to be seen taking firm measures and may cause the authorities to expect trouble. If this is the case, then authorities will start using tougher measures. If the authorities expect malevolent fans, then there is the possibility that the fans will become malevolent and start playing the role after being categorized. Over several rounds, if each player is unhappy about the previous interaction, then they will start to see the other player as increasingly malevolent.

3.2.3 Resource Allocation.

Hypergames are well suited to model resource allocation conflicts. The two applications of hypergames in this area are to water resource managements between multiple players. While hypergames often involve misperceptions, one example shows how different degrees of power over another player can affect the outcome of the game.

Okada, *et al.* first applied hypergame analysis to water resource allocation in Japan's Lake Biwa conflict in the early 1970âĂŹs [266]. The conflict is a water resource management problem, where the downstream users desire more water from

71

the upstream water source, but the controllers of the water source are unresponsive. While each player in the Lake Biwa conflict had misperceptions about the other player's preferences, the hypergame analysis was able to correctly identify the compromise that resolved the conflict historically.

This hypegame has three players: the Shiga Prefecture, downstream prefectures, and the national government. The authors use the notation from Howard [175] and the metagame analysis in [107] to solve the hypergame. While this game is unique in that is models three players, the details are of the analysis are similar to [107].

Hamandawana, *et al.* again applied a game theoretic analysis to a water management conflict [156]. They use a method similar to hypergame analysis to model the interstate conflict between Angola, Botswana, and Namibia over the shared water resource of the Okavango River. The authors use a hypothetical game to build a framework for developing sharing arrangements that minimize conflict, where players make compensatory sacrifices to offset the losses of other players.

Their model introduces the idea of perceived comprised strategic relationships. There are three types: fate control, reflexive control, and behavior control. In fate control, the player's outcome may be influenced by the actions of other players. With reflexive control, the player has some degree of control over the outcome regardless of the actions of other players. Behavior control is the case where the player's outcome is only feasible through interdependent actions of co-partners. This idea follows that of Bennett with perceived games, and Fraser with enforceable/credible equilibriums.

3.2.4 Business.

Hypergames have been applied to business conflicts on many occasions. Business conflicts allow modeling using hypergame naturally, given the misperceptions that arise from company's keeping secrets and leveraging for bargaining position.

3.2.4.1   Applications to Shipping.

Hypergame theory was applied to a conflict in the oil shipping business in [137] [136]. The incident in 1954 almost led to the bankruptcy of Aristotle Onassis, an oil tanker fleet owner. The hypergame analysis showed that decisions made by a player which appear to be irrational under a conventional game theory model, are actually rational when the perceptual limitations and differences in information are considered in hypergame theory.

Hypergame analysis was applied to an ongoing ship building conflict in [44]. The authors were invited by staff of an U.K. shipping company. Ship building had taken off in the 1970's in the U.K., but due to developing countries building completing fleets and the oil crisis in 1973. The hypergame analysis helped to show how different countries supported the crisis in different ways. For example, Japan's profitable industries support the less profitable ones, which allow Japan to keep producing ships when the ship market went into a depression. Other developing countries had labor rates that were below those in the U.K. and support for the ship building industry was lacking in the U.K.

3.2.4.2   Negotiation and Contracting.

Fraser and Hipel explore contract bargaining using hypergame theory [110]. They build a model using the information available to the bargainer and look at the effects of providing opponents with misinformation. They use the model to predict the expected course of events during a negotiation session. The authors provide the first implementation of hypergame analysis on a microprocessor called Conflict Analysis Program (CAP) âĂŞ discussed later.

Fraser and Hipel [111] explore labor-management negotiations, where they apply hypergame analysis to a hypothetical labor-management conflict. The hypothetical conflict is developed in detail in [109]. The authors again use the CAP to show

that the best model does not always conform to the way things should be, but sometimes will conform to how things actually are. For example, they build their model without considering union demands, fairness of salaries, benefits, or working conditions. Instead they model the power of the individual players.

Bennett used a hypergame analysis to explore a conflict where multiple bidders negotiate with a dispenser, who is able to accept the most generous offer [35]. This is a case of two nations bidding to get a multinational corporation to relocate to their jurisdiction. The model focuses on the ability of the dispenser to play bidders against each other.

Graham, *et al.* [144] apply hypergame theory to study supply relationships and modify control systems. They use hypergames to identify misperceptions in the process that are causing inefficiency. These misperceptions are then identified and targeted for correction to improve efficiency in the supply relationships of the players. While the authors are studying twelve pairs of companies, they discuss the types of games created to study the relationship between a vendor of forgings and an engineering company.

3.2.4.3   Trade and E-Commerce.

Stokes and Hipel use hypergame theory to study an international trade dispute over government subsidized export credits [343]. They model the awarding of large contracts to supply subway cars in New York City, which involves the U.S. and Canada, as well as the New York transit authority. Their analysis of the hypergame highlights the role of strategic deception in awarding contracts and presents logically reasonable resolutions.

Hypergame theory is applied to ecommerce by Leclerc and Chaibdraa in [217]. They use hypergame theory as an analysis tool for a multiagent environment. They show how multiple agents interact through communication and a mediator when each

74

has differing views of the conflict. A discussion is also provided on how agents can take advantage of misperceptions.

Novani and Kijima [260] use a symbiotic hypergame model to examine the mutual understanding process between customer expectation and provider capability. They try to formalize the players' internal model dealing with the way each player identifies the situation subjectively and the interpretation function concerning how each player interprets the set of strategies. This model is then applied to different types of customers and providers that the authors develop.

3.2.5 Cyber.

There has been very little research with hypergame theory and its application to cyber warfare, while there has been a variety of research using game theoretic models to improve network security. These models use standard game theory methods, instead of the hypergame theory methods. It is not impossible for hypergame theory to be used where game theory has been applied. If the hypergame model contains valid information, hypergame theory will do as well as the game theory models, with the possibility of outperforming the game theory models [135]. It has also been suggested hypergame agents can be used in place of agents based on decision theory and game theory [179] although no application has been made to cyber warfare defense.

Cybenko [81] studied how different variants of game theory could be applied to cyber adversarial applications. An overview from his work is shown in Table 3.4. He found that hypergame theory provided a high applicability, realism, and robustness, but had a medium level of maturity.

This section describes two research efforts to model cyber warfare scenarios. These are believed to be the only two applications of hypergame theory to cyber warfare and show how hypergame theory is still in its infancy.

3.2.5.1 Information Warfare.

Kopp [198] uses hypergame theory to model Information Warfare. He uses a hypergame to describe how the manipulation of an information channel is reflected in the behavior of the adversaries. Figure 3.9 provides a graphical overview of the general differences between a standard game model and a hypergame model based on information flow. It also shows how hypergames improve upon the game theoretic model by incorporating misperceptions of the players into the game model.

The author focuses on the Information Warfare techniques of denial of information or degradation, deception and corruption, disruption and destruction, and subversion. The hypergame provides a tool for understanding the nature of Information Warfare and allows for quantifying the effects of the action during warfare. The author determines the hypergame theory can be used to model Information Warfare, because the strategies map directly into hypergame models.

3.2.5.2 Model with Obfuscation.

There has been at least a small amount of work performed in the use of hypergame theory applied to cyber warfare. House and Cybenko have a model for generic cyber-attack using hypergame theory [174]. In their model, the defender has the option to choose a specific subgame or the full game, which represents the experience level of the defending administrator. It is based on the HNF work by Vane, using static utility values and placing the attacker as the row player. Learning models are used to determine the belief context percentages representing the possibility that each subgame is being played. The authors ran their simulations for multiple iterations and where able to show the belief context percentages were within $\pm 5\%$ of the true percentages. This research indicates a learning strategy may allow a player to learn the strategy selection of the opponent, thus increasing the maximum utility by obtaining a better understanding of the player's opponent.

76
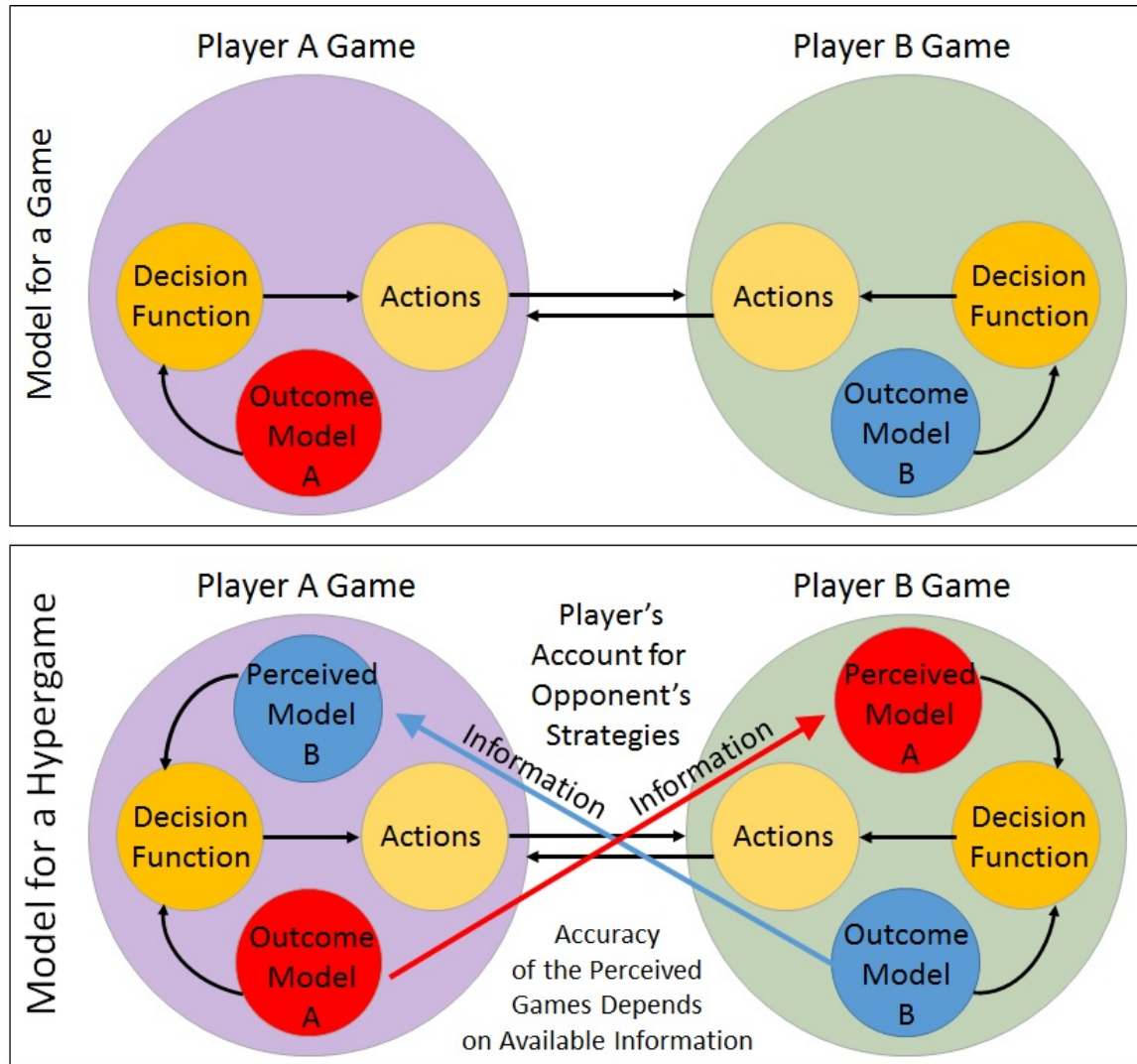
Figure 3.9: Comparison of a game theoretic model and a hypergame model [198].

The authors then look at how the defender can obfuscate the learning ability of the attacker. By using the obfuscation Nash Equilibrium Mixed Strategy, column is able to interfere with row's ability to learn the true percentages. Nash Equilibrium Mixed Strategy is obtained by rearranging the initial utility values used during the row learning experiments. This game setup is contrived in order to allow column to select strategies that will lead to misinterpretation. However, the authors use

this approach because payoffs and subgame definitions are the foundation of the hypergame scenario.

The authors use of hypergame game theory to model a cyber-attack/defense scenario is the most interesting part of this research. Instead of using the full HNF model proposed by Vane, the authors focus on learning the game by repeated play. This is a different approach than proposed originally by Vane [356]. More research into cyber warfare modeling using hypergame theory is needed in order to refine the models and theory.

3.2.5.3   Atacker-Defender Model.

Gibson [135] presents a hypergame model (based on HNF) of the work of Chen and Lenectre [72]. At the heart, this model is an attacker-defender game. It keeps the functional and nonzero-sum utilities from the Chen and Lenectre model.

With Gibson's model, the attacker is given a new strategy, zero-day exploit, which is an attack where there is no defense since the vulnerability is undiscovered. The defender is given two new strategies: providing ruse or shutdown. A defender may provide a ruse by following the attacker into attacking a honeypot, while collecting information about the type and style of the attack. The shutdown option allows the defender to remove the system from the network and stop the attack in its tracks but also removes the system form operation even for mission critical activities. This model is discussed in detail in Appendix B.

3.3   Related Work in Hypergame Theory

This section provides an overview of the related work in hypergame theory. While each of these sections explores expanding hypergames in some way, each is representative of the flexibility of hypergames to be used as models for real-world conflicts.

3.3.1 Hypergame Modeling.

Huxham and Bennett [178] introduce the idea of preliminary problem structuring. In this phase the problem is explored, the relevant participants are identified, along with the possible interactions. The authors try to build up a ' structured picture' in hypergame terms of the situation, instead of a hypergame model. The idea is to explore how the various pieces fit together. The 'structured picture' will often be too complex to form into a formal hypergame model. It is therefore necessary to abstract farther, making simplifications by asking specific questions [178]:

- How two different problem aspects relate?

- Where are the complexities of the system?

- Can simplifications be made while retaining the essential structure?

- Which participants are most important or influential?

In Hipel et al. [166] hypergame theory is applied to modeling misperceptions in bargaining situations. The authors present a new game theoretical model and apply it to a bargaining situation with two or more players (the new model is originally introduced in [361]). They develop a new algorithm, called the HCCAS. The HCCAS algorithm is shown in Figure 3.10.

The real-world situation is represented at the top of the algorithm and provides critical information for the algorithm. The first step is to use the real world information to define the structure of the bargaining situation. This stage involves selecting a point in time at which the analysis will be conducted, as well as identifying the participants, and potential interactions. The second step in HCCAS is modeling, where the actions and outcomes of the players are identified. The third step of HCCAS is the hypergame framework where the bargaining situation structure and

79
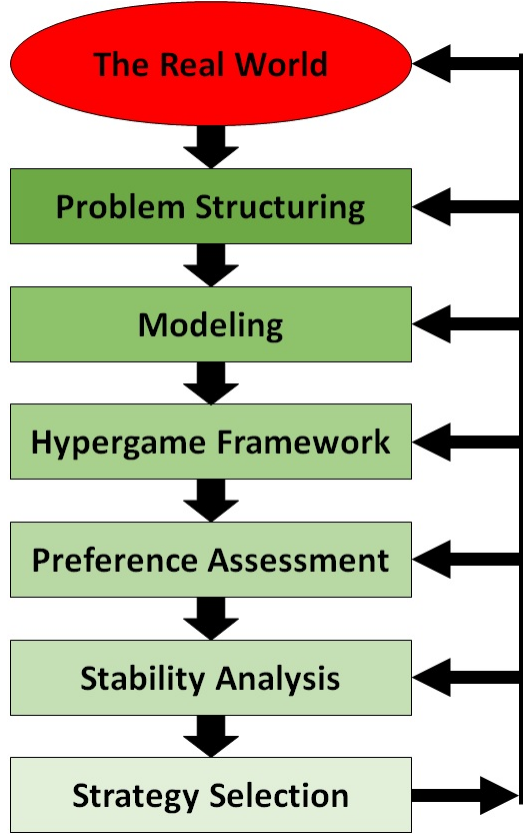
Figure 3.10: HCCAS Algorithm as presented by Hipel et al. [166].

the levels of misperception for each player are identified. Following this step, the preference vectors for each player are formed using information from the previous steps; this is referred to the preference assessment in Figure 3.10. Stability analysis of the hypergame is performed in the fifth step. After this, a strategy is selected and can used to explain the real-world events. The HCCAS algorithm is then applied to the Seymour landfill case, between Eau Claire city and the town of Seymour in Wisconsin.

3.3.2 Stability Analysis.

Wang *et al.* explores stability analysis for n-players in [362]. The authors presents a relationship of possible outcomes, as shown in the Venn Diagram in Figure 3.11.

Nash stability is when players make a rational decision based on the best outcome for the player, this type of outcome is considered rational (R). Nash stability is harder to achieve when misperceptions exist between players. A General Metarational (GMR) outcome is where other players have joint action for player i, and player i cannot achieve a better outcome than the original. A Symmetric Metarational (SMR) outcome is when there is one jointly sequential strategy selection that results in player i achieving the same outcome. If a response to a player's strategy results in that player not achieving a better outcome and the responding player cannot possibly achieve a worse outcome, it is known as a Sequential Stable (FHQ). The contribution of this research is an FHQ outcome exists in all hypergame levels, which implies a GMR outcome also exists in all hypergame levels.
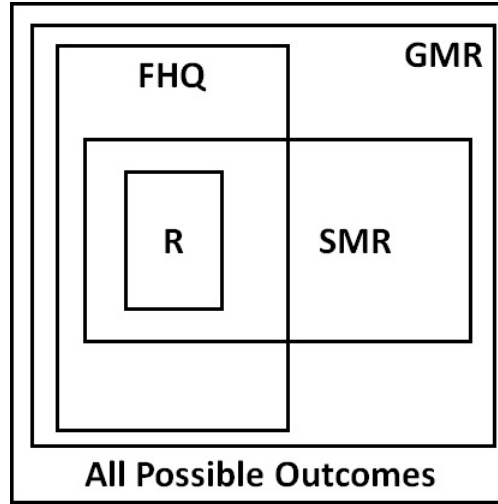
Figure 3.11: Venn Diagram of Stability Analysis Outcomes for n-players [362].

Another view of stability analysis with mixed Strategies is introduced into hypergames by Sasaki et al. [311]. This allows for generalization of Nash's theorem about noncooperative games [252] to hypergames [312]:

Theorem 10. *In every finite hypergame with mixed strategies, there is at least one hyper Nash equilibrium.*

Sasaki proposes the base game as an analysis tool for hypergames [310]. The base game is the overlap of the perceived games between the players [310]. It represents the game that would have been had there been no misperceptions among the players. When a hypergame is compared to the base game, the misperceptions in the hypergame can be analyzed.

Theorem 11. *Let $H = (G^p, G^q)$ be a hypergame with $G^p = (N, \Sigma, u^p)$ and $G^q = (N, \Sigma, u^q)$ where $p, q \in N$. A normal form game $G = (N, \Sigma, u)$ is called the base game of H iff $u_p = u_p^p$ and $u_q = u_q^q$. Let the base game (BG) of hypergame H be denoted by BG_H.*

A Hyper Nash provides an equilibrium solution for a simple hypergame, where a stable hyper Nash equilibrium exists if all the hyper Nash equilibria that exist in a hypergame are also Nash equilibrium in the base game. The hyper Nash equilibrium [312] is used to describe a stationary state, where every player is not willing to change their strategy or perception in the game. It is the solution to a hypergame that is perceived as a Nash equilibrium by every player.

Theorem 12. *Let $H = (G^p, G^q)$ be a hypergame with $G^p = (N, \Sigma, u^p)$ and $G^q = (N, \Sigma, u^q)$. Then $a^* \in \Sigma$ is called a stable hyper Nash (SHN) equilibrium iff $a^* \in N(G^p)$ and $a^* \in N(G^q)$ where N(G) represents the Nash equilibriums for game G.*

This also means that if an outcome is a stable hyper Nash equilibrium, then in the base game it is a Nash equilibrium. This implies that only outcomes that are Nash equilibriums the base game are stationary states in the long run [310]. This gives the following lemma:

Lemma 13. *In a hypergame H, $SHN(H) \subset N(BG_H)$.*

Shown in Figure 3.12, the stability relationships are between the Hypergame (H), Hyper Nash equilibrium (HN) of H (HN(H)),Base Game (BG), Stable hyper Nash equilibrium (SHN) of H (SHN(H).
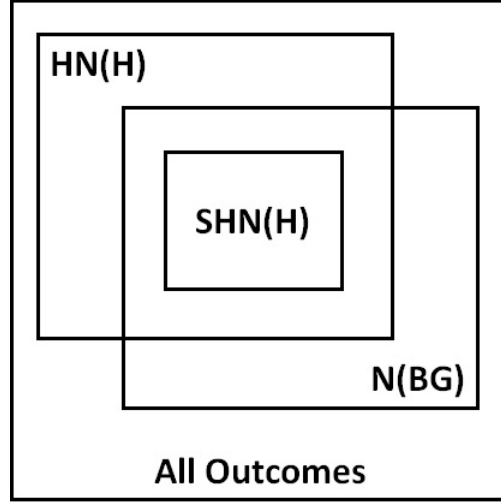
Figure 3.12: Stability Analysis Outcomes for Hyper Nash equilibrium [311].

3.3.3 Player Beliefs.

Vane and Lehner [352] deal with beliefs over games. The hypergame framework allows a player to hedgs its risk about what the other opponents are doing. This is donw by selecting a set of possible game that repreent the action the opponents may take, and then a probability distribution is built over this set of games and evaluated using the maximum expected utility. This allows the player to hedge its risk by using the probably that an opponent will select an action, increasing payoffs by lowering the effect of misperceptions on the hypergame model.

3.3.4 Perceptions and Deception.

Early work in hypergames have used matrices, trees, and tableaux to model interactive decisions [37] [38]. The authors expand this repertoire by showing

83

preliminary problem structuring, where there are games within games (subgames), and build the concept of perception in hypergames. This provides the foundation for using hypergames to solve complex decisions and additional graphical representations of hypergames.

Mateski et al. explores perception, misperception, and deception in conflict using hypergames [235]. They introduce a diagrammatic representation for hypergames called the Hypergame Perception Model (HPM). The HPM is used to model misperception and deception during the Cuban Missile Crisis where perception played a critical role in the conflict. The HPM diagram is shown in Figure 3.13. The two middle columns, denoted Awareness Notation, indicate if the player is aware at the particular level of the hypergame. A check mark indicates correct awareness, and 'XâĂŹ indicates incorrect awareness, and no mark indicates no awareness. Actions available to the players are represented by white circles and strategies are represented by darkened circles.

Player 1		Awareness Notation		Player 1	
Player 1	Player 2			Player 1	Player 2
p_{1_1} p_{1_2} p_{1_3} p_{1_4}	$p`_{2_1}$ $p`_{2_2}$ $p`_{2_3}$ $p`_{2_4}$	✔	✗	$p`_{1_1}$ $p`_{1_2}$ $p`_{1_3}$ $p`_{1_4}$	p_{2_1} p_{2_2} p_{2_3} p_{2_4}
○○○○	○○○○	✔	✗	○○○○	○○○○
○○○○	○○○○	✔	✗	○○○○	○○○○
○○○○	○○○○	✔	✗	○○○○	○○○○

Figure 3.13: Hypergame Perception Model (HPM).

Gharesifard and Cortés [130] present the notion of inconsistent equilibrium in the repeated play of first-level hypergames with two players. Inconsistent equilibrium

refers to the equilibria of the hypergame where at least one player expects the other to move away from. Just the existence of inconsistent equilibrium means there is some misperception about the game among one of the players. A class of actions, call exploratory, are also identified by the authors to allow players to move away from inconsistent equilbiria and decrease the misperception. If only one player in the game uses exploratory actions, then the hypergame will arrive at an outcome rational for the player. If both players use exploratory actions, then the repeated play may finish in a cycle.

They [132] also study the situations where the perceptions of players in the game are inconsistent and evolving. The authors present a new method, called swap learning, which allows the incorporation of information gained by observing their opponents actions into the player's beliefs. This method allows a player to decrease misperceptions, but at a cost of incorporating inconsistencies into their beliefs. For example, if player A originally believes player B's preferences are 15 >3 >7 >11 (where each digit is an outcome), but the player B's actions leads from outcome 15 to 11, then player A interchanges the positions of outcomes 15 and 11. Player A would then believe player B's preferences are 11 >3 >7 >15; this is called swap learning. Since the swap of preferences does not take into account the other outcomes, then inconsistencies can form in the beliefs of player A. To eliminate the inconsistencies, the modified swap learning method is presented. This method assumes that the opponent has perfect information and plays their best strategy, buts yields consistent beliefs and decreases player misperception. The swap learning method place the origin of the misperception on the player performing the belief update.

Again, Gharesifard and Cortes [131] [133] focus on conflicts with incomplete information, where players may have different perceptions about the conflict. Specifically they focus on a 2-player hypergame where one player, the deceiver, has

full information about his opponent's game and wants to introduce a certain belief in it. They use their previously developed H-digraph [128], a special class of digraph used to encode the belief structure of the hypergame players. Using the H-digraph they are able to characterize deception when stealthy actions are possible in the game. Their papers [128] [129] [132] also presents two algorithms for updating perception in the hypergame. These methods can decrease the misperception between the player's perceived game and true payoffs.

3.3.5 Dynamic Payoff Functions.

Gibson presents a model based on the intrusion model presented by Chen and Leneutre [135] and the Hypergame Normal Form model presented by Vane [355, 356]. Appendix B contains a detailed discussion. The author achieves a model that has a changeable nonzero-sum utility values with a process for delineation of strategy selection [72]. In order to achieve this model, the Chen and Leneutre intrusion model is extended by adding strategies for both the attacker and defender, while the HNF model is used to hide r discount strategies from the other player.

3.3.6 Mutual Interaction.

Inohara *et al.* discuss the ability of players to engage in multiple games simultaneously [182]. Each game a player engages in may have interactions with other games which can affect outcomes. The basic example they give, is a situation in which a company competes in two different markets with two different opponents (i.e. in market Y, Company A competes with Company B and in market Z, Company A competes with Company C). A similar real life conflict would be a global company deciding whether to invest in a specific country's market knowing they will be face completion from the country's local established vendor. They integrate different games in order to capture the interactions, which is realistic of real-life situations, and

an example is given using the hypergame methodology, in order to model hypergames that are mutually interactive and increase perception ability of players.

3.3.7 Fuzzy Logic.

Song *et al.* [338] [337] present a novel method that uses fuzzy logic to obtain the outcome preference in first-level hypergame models. A fuzzy aggregate algorithm is applied to get the group fuzzy perception of the opponents ' outcome preference. The preference sets are then obtained by solving linear programming models. The authors obtain the crisp perception for the opponents' outcome preference by using a defuzzification function and the Newton-Cotes numerical integration formula. The authors then use the concept of consensus winner to determine the preference vectors in the hypergame models. In [339], artificial neural networks (ANNs) are trained to learn the criteria for comparing fuzzy outcome preference numbers.

Yong *et al.* [291] use fuzzy pattern recognition to establish a nonlinear programming model. This model is used to integrate different outcome preferences for opponents perceived by different experts. Each expert perceives the outcome of the game and this information is processed using fuzzy pattern recognition to obtain a standard outcome.

Zeng *et al.* [371] develop an integration model for hypergames with fuzzy preference perceptions. In conflicts, players cannot perceive information about the opponent's game clearly, so an integration model of multiple perceived fuzzy games, using hypergames is developed. Each player has fuzzy preference perceptions. The authors use linguistic values for the outcome preferences over the outcome space, which are represented as triangular fuzzy numbers. Hypergames with fuzzy preference perceptions are demonstrated with a military example about two country's Navys.

3.3.8   Bayesian Games.

Yasuo Sasaki and Kyoichi Kijima compare Bayesian games with Hypergame games [313]. A detailed analysis of their work appears in Appendix D. The discussion includes what was accomplished in their research, how the claim that Hypergames can be reformulated in terms of Bayesian games is stronger than the method they actually propose, and covers the uniquess of hypergame as proposed by P.G. Bennett [34] and later refined by Russell Vane [356].

3.3.9   Multi-agent Environments.

Chaib-draa [68] use hypergames to analyze differences in perceptions in multi-agent environments. They show how multi-agents can interact using a third party, while having different views and perceptions of the game. The third party is used to observe the exact perceptions of the players from an external context. The players can then choose to trust the external observation and update their perceptions of the game (with assurance from the third party of correct perception). If one of the players deviates from the agreed upon outcome, the third party informs the other player. Overall, the third party is used to enforce perceptions between players or to create misperceptions between players. For example, the third party could be used to have nested perceptions in different hypergame levels.

3.3.10   Combining Approaches.

Huxham and Bennett [177] explore combining hypergames with cognitive mapping, since both deal with the subjective world of decision-makers. They started with the idea that maps could be built up, then the players, preferences, and outcomes could be extracted. The authors determined this process was not straightforward. They then structure the problem in hypergame form then used piecemeal maps to explore certain outcomes. The relationship between hypergames and cognitive mapping is explored theoretically by Bryant [59].

Bennett and Cropper [33] examine combining hypergames with Strategic Choice to provide an effective method for modeling decision problems. Strategic Choice deals with uncertainty [113], where a participant moves between the activities of: problem-shaping, generating alternatives, comparing solutions, and finally choosing how to act. While hypergames and Strategic Choice often deal with uncertainty, both offer different perspectives. In Strategic Choice, the emphasis is on the need to coordination between parties, where in hypergames the emphasis is on communication as a means to makes threats, bluffs, or deception [33].

Putro *et al.* [290] [289] [288] combine hypergames with genetic algorithms to produce adaptive learning procedures. The genetic algorithm is used to choose nature's strategies in order to improve perceptions. They present three learning methods where each method varies a part of the genetic algorithm (such as fitness evaluation, modified crossover, action choice). The authors present two experiments that analyze the effect of uncertainty and crossover rates on the outcome of the learning procedures.

Kanazawa *et al.* [191] [190] [192] study hypergames and evolutionary game theory. They use hypergames to add perceptions to evolutionary game theory, which result in evolutionary hypergames. Interpretation functions, which specifies the relationship between the player's strategies and those of their opponent(s), from hypergames are introduced into evolutionary games. These interpretation functions are then used to create the replicator dynamics for the evolutionary game, which describe the selection process for the distribution of the strategies in a given population. This process is demonstrated using the original application by Bennett to Soccer Hooliganism [190].

3.3.11 LG Hypergames.

While not directly related to hypergame theory as envisioned by P.G. Bennett, LG Hypergames have a similar goal: to "account for drastic mutual influence of multiple subgames" and are applied to abstract board games (ASB) [341] Linguistic Geometry (LG) hypergame was first demonstrated in [341], where it was used to infer the direct and indirect effects. Each ASB is dynamically linked together by interlinking maps, a concept similar to hyperlinks in an HTML document [342]. A detailed application of LG hypergames is given in [363].

3.3.12 Conway Games and Hypergames.

Honsell and Lenisa study Conway games, forming a "hypergame" from basic Conway games [173]. A Conway game is a combinatorial game with 2-players, no chance, a set of positions for both players, and perfect information [78]. These games represent board games such as Nim or Go. Games where both players have the same set of moves (Nim) are called impartial and games where players have different sets of moves (i.e. Go) are called partizan [173]. In this case the term hypergame refers to a non-wellfounded game or a game that do not terminate. The term is not used in the same way as this research.

3.4 Hypergame Analysis Software

Hypergame analysis is possible by hand but not recommended; it is a tedious process which is better accomplished by software using the computational power of modern computers. Hypergame analysis requires calculation of utility from mathematical functions, multiple runs of game models with different strategy selection such as static or random, as well as update model variables and player belief contexts between each game iteration. The ability of software to fit these requirements are discuss as different tools are explored for hypergame analysis.

3.4.1   Statistical Software Packages.

Microsoft Excel, MiniTab, and other statistical software packages are able to model hypergames and can easily calculate utility functions from mathematical equations as shown at the IEEE/WINFORMS Joint Program for Capital Science [267]. The main disadvantage is these programs have the inability to run multiple game iterations and update variables between iterations. Mathematical software, such as Matlab or Mathematica, can calculate the utility functions from mathematical equations and run multiple game iterations and update variables as well as player beliefs between iterations, but this software is not specialized for hypergame analysis. This means for each game model the entire model has to be built from scratch; there is no standardization of hypergames between researchers.

3.4.2   Gambit.

Gambit is software designed for analyzing finite, non-cooperative games using the strategic form [239]. Players and strategies can be added using the Gambit interface to quickly create a game for analysis. It has the ability to exchange game model to external tools, creating a standard for game theory model data. The main disadvantage of this software is lack of support for the complex hypergame model; there is no way in the Gambit interface to enter different games based on each player's perceptions or to use mathematical equations to calculate utility values during game analysis.

3.4.3   HYPANT.

A software tool specifically designed for hypergame analysis, called HYPANT, was written by Lachlan Brumley [58]. It uses a standard notation, referred to as a language, to represent hypergame models called Hypergame Markup Language (HML). The HML allows the hypergame model data to be saved, restored, and transported, as well as supported subgames based on the player's perceptions. The

disadvantages to HYPANT are the lack of support for functional utility values, it only supports the stability and unilateral improvement values used by Frasier and Hipel in their analysis of the Cuban Missile Crisis [106].

3.4.4 SPA.

Another hypergame analysis program based on Vane's HNF theory is called Security Policy Assistant (SPA). SPA was created to assist in deciding if classified documents are released or withheld from foreign disclosure [183]. While the software manual was available, the software is not given its sensitive nature in decision making with classified information. This software supports the application of hypergame theory beyond the previous applications of military, sports, and business conflicts, given this software's ability to assist in decision making about classified documents.

3.4.5 HAT.

The lack of suitable software meeting all the requirements for hypergame analysis caused Alan Gibson to create the HNF Analysis Tool (HAT) software [135]. HAT is written in Java and supports using the Extensible Markup Language (XML) to input and save game design. XML is an improvement over the HML language used by HYPANT because XML is widely supported, has many tools to create, read, and verify, as well it is not proprietary like HML.

Once a game in XML is loaded, HAT allows multiple game iterations to be run, supporting static or random strategy selection. It also allows variables and belief contexts to be updated between hypergame iterations. Given the availability of the HAT software and its ability to handle hypergames using HNF concepts developed by Vane [356], this software is used and updated throughout this research effort. The HAT software is shown in Figure 3.14 with a game loaded.

					CΣ	0.9351	0.0649	0.0
	0.8				C1	0.9351	0.0649	-
		0.2			C0	0.9351	0.0649	0.0
WS	PS1	MO	R1	R0=FullGame	Defender \ Attac...	Not Attack	Attack	Zero-Day Exploit
0.5	0.5	1.0	0.5	0.5	Not Defend	0.0, 0.0	-1.0, 0.9	-1.0, -1.0
0.5	0.5	0.0	0.5	0.5	Defend	-0.118, 0.0	0.7, -0.9	-1.0, -1.0
0.0	0.0	0.0	0.0	0.0	Provide Ruse	-2.0, 0.0	-1.0, 0.9	-1.0, -1.0
0.0	0.0	0.0	0.0	0.0	Shut Down	-3.0, 0.0	-3.0, -1.0	-2.0, -1.0
-0.0649	-0.0649	-0.0649	-0.0649	-0.0649	<-- EU			
-1.0	-1.0	-1.0	-1.0	-0.0649	<-- G			
-0.2519	-0.2519	-0.2519	-0.2519	-0.0649	<-- HEU			
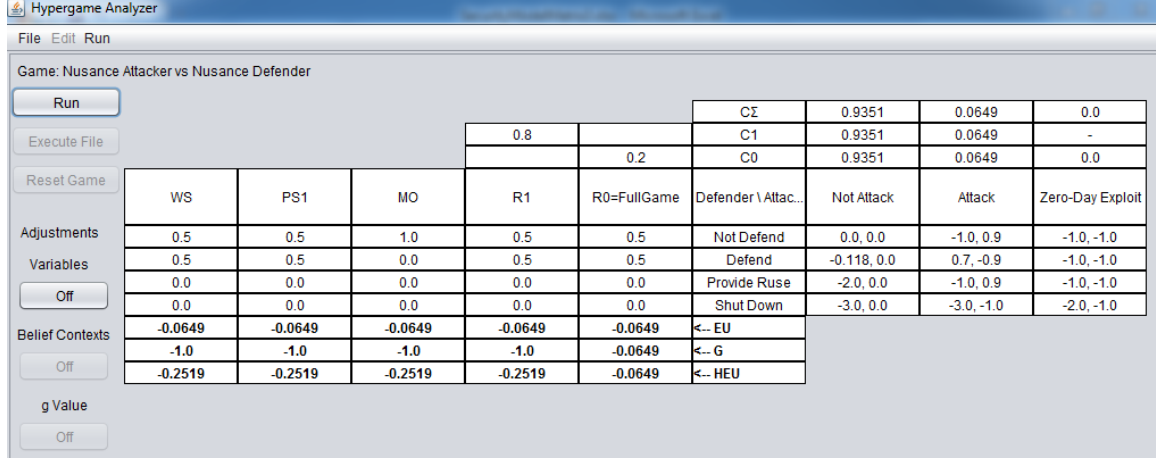
Figure 3.14: HNF Analysis Tool (HAT) Software.

3.4.5.1 Nash Equilibriums.

The HAT software uses the Lemke-Howson algorithm to calculate Nash equilibriums which is appropriate for non-zero sum bimatrix games [218]. This algorithm is not guaranteed to find all Nash equilibriums but will find at least one as proven by Nash's Existence Theorem [252]. This Nash Equilibrium is then used as the initial belief context. Additional belief contexts are used for the creation of hyperstrategies.

3.4.5.2 Belief Contexts.

Belief contexts consist of row's belief that column will choose the specific context and the percentages of each strategy column can choose from. Each belief context forms a subgame where column strategies with a percentage use of zero are removed. Subgames are determined using the supplied belief contexts in the XML game file and by removing all strategies labeled as hidden. This allows hyperstrategies to be created by examining the subgames from the hypergame analysis.

The HAT software generates the hyperstrategies as the Modeling Opponent (MO), Pick Subgame (PS), and Weighted Subgame (WS). These subgames are not actually hyperstratgies, but are included by the software for a complete model.

3.4.5.3 Utility.

The Expected Utility (EU), worst case utility (G), and Hypergame Expected Utility(HEU) are calculated for each hyperstrategy. The EU is calculated by multiplication of utility values with row's strategy selection percentage, as well as the aggregate of column strategy selection in C_{Σ}. The final EU value is calculated by adding each strategy utility together. The G value is calculated using the worst case outcome, where the lowest utility available to row's strategies is multiplied by the strategy's percentage use. The HEU is calculated as follows:

$$HEU(hs) = EU(hs) - (EU(hs) - G(hs)) * g$$

The fear-of-being-outguessed, g, is used to calculate the HEU of each hyperstrategy. The HAT software allows the g value to be fixed (predetermined) or changed between executions so each game can have differing HEU values. Different HEU values can lead to different hyperstrategy choices.

3.4.5.4 Game Execution.

Each hypergame can be executed, which consists of selecting player strategies and calculation of utility values. The strategies for the column player are selected by a usage value or usage file. Usage values are in the XML game file as percentages assigned to each column strategy. The percentages are then used to choose a column strategy stochastically. If a usage file is used, then the file contains a list of strategy names and the strategies are chosen in order. The strategy for the row player is determined by selection of a hyperstrategy. The strategy is selected from the hyperstrategy with the highest HEU value and a random number. Based on the strategy selected by row and column results in the utility each player receives in the

game outcome. The results of the execute game can be exported to a file in comma separated value (CSV) format for manipulation by external software.

3.4.5.5   Update Modes.

The HAT software supports for update modes for game execution. The variables, g value, and belief context can be updated. The fourth mode is where all are updated. Variables are updated using a supplied algorithm which can be affected by strategy choices. This allows costs for certain actions to change over time. The update algorithms are contained in the XML file variable node. The g value is updated by determining if the player was able to obtain the expected utility for that particular game iteration. If the expected utility is reached then the g value decreases, otherwise it increases. The values to increase or decrease by are settable within the XML file. The initial belief context are based on utility values, which are affected by the variable changes. If the game changes, then it is expected the player's belief about how the game is being played will change. The belief context will be updated whenever the utility values are changed in the software.

3.5   Summary

This chapter describes the foundational model for hypergames, as well as provides applications of hypergames that indicate its ability to model complex conflicts with imperfect information. The foundation provided in this chapter is built upon in the following chapters as hypergame theory is extended with temporal logic. The next chapter provides an overview of temporal logic, which is used in this research to extend the hypergame model.

Table 3.3: Listing of Hypergame Applications, Chronological.

1st Author and Citation	Year	Category
Giesen [136]	1978	Business
Bennett [39, 40]	1979	Military conflicts
Giesen [137]	1979	Business
Wright [369], Shupe [328]	1980	Military conflicts
Bennett [42]	1980	Sports
Fraser [109, 110]	1980	Business
Bennett [35]	1980	Business
Bennett [44]	1981	Business
Fraser [111]	1981	Business
Said [308]	1982	Military conflicts
Bennett [41]	1982	Military conflicts
Fraser [105, 107]	1983	Military conflicts
Stokes [343]	1983	Military conflicts
Okada [266]	1985	Resource allocation
Hipel [167]	1988	Military conflicts
Graham [144]	1992	Business
Vane [355]	1999	Cyber
Vane [356]	2000	Cyber
Leclerc [217]	2002	Business
Kopp [198]	2002	Cyber
Hamandawana [156]	2007	Resource allocation
Novani [260]	2010	Business
House [174]	2010	Cyber
Gibson [135]	2013	Cyber

Table 3.4: Variants of Game Theory and applicability to Cyber Adversarial Applications

	Maturity	Applicability, Realism	Robustness	Preliminary Results
Games with Complete Information	high	low	low	low
Games with Incomplete or Imperfect State Information	high	high	medium	medium
Games with Incomplete or Imperfect Objective Information	high	medium	medium	low
Adaptation, Evolution, Learning in Games	medium	high	medium	medium
Hypergame Theory	medium	high	high	high
Behaviorial Models	medium	high	medium	medium

IV. Temporal Preliminaries

This chapter presents the background and related work for temporal logic, logical reasoning, and belief revision. First, background information on the foundation of temporal logic is given. Then logical reasoning, such as deductive, inductive, and abductive reasoning is discussed. Finally, a brief overview of belief revision is given with application to game theory.

4.1 Temporal Logic

Until now, past research using hypergame theory has not considered time as an integral part of the hypergame model, this research uses temporal logic to represent the changes to the hypergame model as player perceptions change over time. The only known research to use iterations with hypergame theory was in the analysis of the Falkland/Malvinas conflict [167]. Here the authors picked three distinct points in time and created three individual hypergames to model the conflict.

Temporal logic provides a method and notation to impose constraints on a time based model. By using temporal logic to constrain the hypergame model, time is able to be incorporated leading to more accurate modeling of real life conflicts. Real life does not happen in distinct iterations, instead events play out over time with information and beliefs being updated overtime.

4.1.1 Temporal Logic History.

Modal logic is based on the notion of necessity and possibility. Introductory writing on modal logic can be found in Hughes and Cresswell [176] and Chellas [71]. It was developed by philosophers to understand the different modes of truth. It flourished during Medieval times when it was used for theological argumentation

[229]. Widely accepted semantics for modal and temporal logic were developed by Kripke [206].

On one hand, temporal logic is believed to have evolved from modal logic, through the process of interpreting the modal operators in the context of time-dependency. Alternatively, the logic can be specialized with time modalities. Rescher and Urquhart [300] and Goldblatt [139] study this view in detail.

Logical analysis of natural languages provide additional motivation for the study of temporal logic This view evolves temporal logic form the formalization of linguistic conventions where tenses are modeled using formal calculus. The seminal paper on this view was published by McTaggart [240]. Prior further applies this approach in [287], along with Kamp [189], Gabbay [117], and van Benthem [45].

Pnueli [283], Goldblatt [139], and Emerson [94] provide general surveys on temporal logic uses in Computer Science, while Fisher et al. [102] presents a survey on temporal reasoning in Artificial Intelligence.

4.1.2 Types.

Temporal Logic Temporal Logic (TL) for reasoning about concurrent programs can be divided into different types: propositional versus first-order, global versus compositional, branching versus linear, points versus intervals, and past versus future tense. The various types are discussed in more detail below.

4.1.2.1 Propositional versus First-order.

Propositional TL is based on non-temporal classical propositional logic. The proposition is built by a formula of atomic propositions, which are used to express atomic facts about the concurrent system state, truth-functional connectives, such as $\wedge$, $\vee$, $\neg$ (and, or, and not), as well as temporal operators. Wolper [367] presented an extension to propositional temporal logic with right-linear grammar operators, showing the resulting system leads to greater expressive power. Banieqbal

and Barringer [24] show Wolper's proof is complete with some modifications. An alternative is propose by Wolper et al. [367] where finite automata are used on infinite words.

Atomic propositions are refined into expressions, created from variables, constants, functions, predicates, and quantifiers. These expressions are referred to as First-order TL. There are many sub types of First-order TL: uninterpreted, interpreted, fully interpreted, and partially interpreted. In uninterpreted First-order TL, no assumptions are considered about the special properties of the structures, while interpreted TL makes assumptions about a specific structure. In fully interpreted First-order TL, each variable has a specific domain and each function symbol has a concrete function over the domain. On the other hand, when a specific domain is assumed but the function symbols are uninterpreted, this is called partially interpreted First-order TL. TL as distinguished between local and global variables. Local variables are assigned values in different states, where the value can differ between states. Global variables are assigned a single value, which holds over all states of the system.

Syntactic restrictions can be imposed on the interaction of temporal operators and quantifiers. In unrestricted syntax, temporal operators appear in the scope of the quantifiers and are normally undecidable. Restricted First-order TL does not allow temporal operators to appear in the scope of the quantifiers. This results in a propositional TL and a first-order language for stating atomic propositions.

4.1.2.2   Global versus Compositional.

Endogenous TL interprets all temporal operators corresponding to a single concurrent program, in a single universe. Exogenous TL allows the temporal operators to express the correctness properties related to program fragments or different programs within the same formula. This allows compositional reasoning, where

the whole program is verified by specifying and verifying each of the subprograms, then combining each subprogram to obtain the proof of correctness for the complete program. In this case, each sub game proof is used as a lemma.

4.1.2.3 *Branching versus Linear Time.*

Two possible views of the nature of time exist when defining a temporal logic system. If at any given moment there is only one possible future moment, time is linear. If at any given moment there are alternate courses leading to different possible future moments, time is branching. Linear time logic is used to describe events over a single time line, while branching time logic allows quantification over possible futures. These views have been used to reason about programs in [213] [96] [282].

4.1.2.4 *Points versus Intervals.*

Most program reasoning use temporal operators that evaluate to true or false at a certain point in time. Temporal operators can also be evaluated over intervals of time. Intervals of time have been used by [317] [247] [155], and are claimed to simplify the creation of specific correctness properties.

4.1.2.5 *Discrete versus Continuous.*

Time is discrete if the present moment corresponds to a program's current state and the next future moment corresponds to the program's next successor state. Time progresses in discrete units using nonnegative integer values. Discrete time has been applied to real-time system in Koymans et al. [203], Ostroff [271], Alur and Henzinger [17] [18], Harel et al. [158], and Henzinger et al [163]. Time is continuous if the current state of the program and the successor state are continuous, for example over the reals or rationals instead of only at distinct values. Burgess [60] offer a proof for temporal logic over continuous time domain, and Burgess and Gurevich [61] analyze the decidability of the satisfiability problem. Barringer et al. [29], Koymans and de

Roever [202], and Alur et al. [16] apply continuous-time temporal logic to real-time systems.

4.1.2.6   Past versus Future.

Classical temporal logic defined by Kamp [189] includes both past and future operators. The future tense operators are used to describe what may happen to future system states, while past operators allow the system to take into account what happen in the past. Past tense operators are similar to history variables and allow for compositional specification. Gabbay et al. [121] claim the expressive power is not reduced by restricting the logic to only future operators. On the other hand, Lichtenstein et al. [222] and Pnueli [283] show that the past operators lead to more uniform classification of program properties.

4.1.3   Common Language.

4.1.3.1   Model.

The general model for point-based temporal logic is (S, R, π), where S is the set of time points, π maps each point to true propositions at the given point in time, and R is an earlier-later relation between the points in S [103]. In discrete temporal logic, the accessibility relation, R, can be replaced by a relation between adjacent points, N. This forms the *next-time* relation, which is applied over the set of all discrete moments in S. Therefore, for all s_1 and s_2 in S, N(s_1, s_2) is true if s_2 is the *next*discrete time moment after s_1.

For non-discrete models, such as the reals, $\mathbb{R}$, there is no clear idea of the *next* point in time. If a temporal relation, R, is based on $\mathbb{R}$ then if the two time points are related, there is always another time point between the points, since $\mathbb{R}$ is *dense* [103]:

$$\forall i \in S, \forall k \in S. R(i, k) \Rightarrow [\exists j \in S, R(i, j) \wedge R(j, k)]$$

The *next* point in time does not make sense in this context, so logic based on dense models use operators relating to intervals. These require interval-like operators, which refer to a particular subsequence of points.

4.1.3.2   Representation.

In temporal logic, temporal operators are used to reason about how truth values vary with time. Two temporal operators are *sometimes* P which is true now if in the future there is a time moment at which P becomes true and *always* Q which is true now if Q is always true at all future time moments.

□φ - φ is *always* true in the future

◇φ - φ is true at *some time* in the future

There exists temporal aspects that cannot be represented using □ and ◇ [189] [51] [367]. Kamp [189] and Burgess [60] introduced the *until* operator, : $\mathcal{U}$, and the *unless* operator, $\mathcal{W}$, from tense logic:

φ$\mathcal{U}$ψ - there exists a moment when ψ is true and φ will continuously be true from now *until* this moment

φ$\mathcal{W}$ψ - φ will continuously be true from now on unless ψ occurs, at which time φ will cease

The *unless* operator is often referred to as a *weak until*, because of the connective similarities. In most situations this is fine, since *sometime* and *always* can be defined using the *until* operator. The *next time* operator, ○, is added as a convenience in discrete models of time.

○φ - φ is true at the *next* moment in time

Using the *next-time* relation, in the discrete case, example semantics can be given over model M = ⟨S, N, π⟩:

⟨M, s⟩⊨ ○φ iff, ∀ t ∈ S, if N(s,t) then ⟨M,t⟩⊨ φ

In some cases it is possible to define the ○operator directly using the $\mathcal{U}$ operator [95].

Past-time connectives, such as *since*, were originally incorporated in tense logics [189] [60]. These connectives were originally not used by temporal logics in Computer Science, but have been added for convenience [28] [222]. If both past and future operators are required in a temporal model, are a matter of discussion [216]. Therefore there are past-time counterparts of □, ◇, etc., such as the *previous* operator, ●, the past-time dual of the *next* operator.

● φ - φ is true at the *previous* moment in time

A more general definition, dependent only on the discreteness of the model, shows the interaction between the two operators. The *next-time* relation is used and ● is defined over model M = ⟨S, N, π⟩:

⟨M, s⟩⊨ ○φ iff, ∀ t ∈ S, if N(s,t) then ⟨M,t⟩⊨ φ

⟨M, t⟩⊨ ●φ iff, ∀ s ∈ S, if N(s,t) then ⟨M,s⟩⊨ φ

The duality between ● and ○is seen with ●false (or ○false). ●false can only be satisfied in a temporal model at the first or last moments. Using the previous definition the only way ●false is satisfied is if there are no previous moments in time. For example, given the axiom:

φ ↔ ●○φ

The state, s, is either disallowed by the axiom, or if the state is allowed then it cannot be distinguished from the current state in any temporal formula. The

interactions of the *until* and *since* or the *sometime future* and *sometime past* are explored in [123] [349] [303].

For example, in Computer Science and Artificial intelligence, first-order logic statements are used in temporal modeling. This treats one of the arguments to each of the predicates as a time parameter and is called the temporal arguments approach. Given that each statement is to be evaluated at the moment in time i, the following formulas can be represented in classical logic [103]:

$$p \land \bigcirc q \longrightarrow p(i) \land q(i+1)$$

$$\diamond r \longrightarrow \exists j, j \geq i \land r(j)$$

$$\square s \longrightarrow \forall k, k \geq i \rightarrow s(k)$$

The *until* and *since*, as well as *sometime in the future* and *sometime in the past* are useful for linear models, not only discrete models [189]. These operators describe temporal properties in dense, non-discrete models. *Until* and *since* have been used to change arbitrary formulas into normal form, this allows past-time to be separated from future-time [27] [101] [118] [170]. While *sometime in the future*, F, and *sometime in the past*, P, have been used in non-discrete logics based on $\mathbb{R}$ [122] [123] [119].

In [29] and [194] *until* is the basic temporal operator and the temporal model is based on $\mathbb{R}$. This allows only the future moments in time to be considered. The authors had trouble with the model over $\mathbb{R}$, so they introduced the additional constraint of finite variability. In finite variability, a property's value is allowed to only change a finite number of times, between two points in time. This prohibits the property from varying between true and false infinitely over a fixed, finite period of time. Finite variability is also used in [83] [126].

4.1.3.3 Interval Temporal Representations.

Two different approaches exist for interval temporal representations. Allen developed interval algebra for Artificial Intelligence, while Moszkowski et al. developed interval temporal logic for Computer Science. Thses two approaches are discussed below.

Allen Interval Algebra is referred to as the reification approach. In this approach, predicates like *holds* and *occurs* are applied to properties and moments in time and each property will either hold or occur. Allen used binary relations for intervals to describe the relationships of intervals [12] [13]. For example, for interval I_k where k is an specific time period:

I_A *overlaps* I_B is true if the intervals I_A and I_B overlap

I_A *during* I_B is true if the interval I_A is contained completely in the interval I_B

I_A *before* I_B is true if the interval I_A occurs before I_B

Allen Interval Algebra was further formalized and analyzed in [224], [223], [212], [211], [154], [14]. The algebraic properties are explored in [169] [168]. Allen's initial binary relations have been extended in [147] and associated with computational problems in [89]. There are also other publications on reified approaches, such as McDermoott's logic of plans [238], Situation Calculus [298] [225], and the Event Calculus [201]. Detailed surveys on reified approaches can be found in [227], [125], [25], and [296].

Moszkowski et al. [153] [245] developed interval logic that was directly related to the discrete propositional temporal logic [120]. Moszkowski et al. originally developed their logic to model digital circuits and called it ITL.Formulas in ITL are interpreted in sub-sequences $(\sigma_b, \ldots, \sigma_e)$ instead of at a specific point in the model σ. This means the propositions are only evaluated at the start of the interval. Given proposition P:

⟨σ_b, . . ., σ_e ⟩$\models$ P iff, P $\in \sigma_b$

With this definition, the definitions of *sometimes* and *next time* are as follows:

⟨σ_b, . . ., σ_e ⟩$\models \Box \varphi$ iff, $\forall$ i, if b $\leq$ i $\leq$ e then ⟨σ_b, . . ., σ_e ⟩$\models \varphi$

⟨σ_b, . . ., σ_e ⟩$\models \bigcirc \varphi$ iff, e $>$b and ⟨σ_b, . . ., σ_e ⟩$\models \varphi$

ITL uses the *chop* operator, ;, to fuse time intervals together [305] [359] with the basic temporal operators:

⟨σ_b, . . ., σ_e ⟩$\models \varphi$; ψ iff, $\exists$ i such that b $\leq$ i $\leq$ e where both ⟨σ_b, . . ., σ_e ⟩$\models \varphi$ and ⟨σ_b, . . ., σ_e ⟩$\models \psi$

The *chop* operator is problematic because it guarantees a high complexity logic [103]. It is highly useful because it allows the splitting of intervals on their properties. For example, if there is a sub-interval where true is satisfied which is immediately followed by a sub-interval where φ is satisfied:

$\diamond \varphi \equiv$ true; φ

Further examples of formulas in ITL are given with explanations [103]:

- p persists through the current interval

    $\Box$ p

- Definition of steps within an interval

    up $\wedge \bigcirc down \wedge \bigcirc\bigcirc up \wedge \bigcirc\bigcirc\bigcirc down$

- Sequences of intervals can be constructed

    $\Box$ January; $\bigcirc\Box$*February*; $\bigcirc\Box$*March*; . . . .

- p has a period of being false which is followed by a period where it is true

$$\Box\neg p;\bigcirc\Box p$$

Granularity in ITL has been explored with the temporal projection operator [57], [149], [150], and [246]. Halpern and Shoham have proposed HS logic over intervals [154]. This logic captures Allen's algebra with unary modal operators and uses binary operators to capture the *chop* operator from ITL [141].

There are several extensions to the interval approaches already discussed. In [124] and [75] intervals over arbitrary relations are considered, this allows spatial and spatio-temporal logics. Spatial logic is any formal language interpreted over geometical entities and relations [5], while spatio-temporal logic is a combination of spatial logic and temporal logic. Duration calculi [373] [70] on been used to introduce real-time concepts into interval temporal logics. An extension to interval temporal logic that allows endpoints to be moved called compass logic is explored in [233].

4.1.3.4   Quantification.

Propositional temporal logic has been extended to allow quantification over propositions. This allows first-order quantifier symbols, $\forall$ and $\exists$ to be used with boolean valued variables. This is referred to logic Quantified Propositional Temporal Logic (QPTL). For example, the following formula is allowed:

$$\exists\, p,\ p \wedge \Diamond\, \Box\, \neg p \wedge \bigcirc\bigcirc p$$

This form of quantification is called substitution interpretation [152] and is defined as:

⟨M, s ⟩$\models \exists$ p. φ iff, there exists a model M' such that ⟨M, s ⟩$\models \varphi$ and M' differs from M at most by the valuation given to p

108

Substitution interpretation is used in QPTL and in other extensions to PTL. Haack [152] presents a discussion on the differences between substitution interpretation and the more common objectual interpretation of quantification:

⟨M, s ⟩⊨ ∃ p. φ iff, there exists a proposition q ∈ PROPERTIES such that ⟨M, s ⟩⊨ φ (p/q)

where φ (p/q) is the formula φ with p replaced by q throught the formula

QPTL allows regular properties to be defined, and was inspired by Wolper's work [367] right-linear grammar operators for PTL, called ETL. These right-linear grammar operators are restricted fixpoint operators [229]. Wolper [52] also introduced the notion of fixpoints, extending PTL with *least*, μ, and *greatest*, , as fixpoint operators. This allowed expression of more complex expressions:

□ φ ≡ . φ ∧ ◯ξ

Where □ φ is the maximal fixpoint, ξ, of the formula $\xi \rightarrow (\varphi \wedge \bigcirc \xi)$. Therefore, the maximal fixpoint results in □ φ as the infinite formula:

φ ∧ ◯φ ∧ ◯◯φ ∧ ◯◯◯φ ∧ ◯◯◯◯φ ∧ . . .

Related work has shown the extensions QPTL, ETL, and fixpoint are expressively equivalent under specific circumstances [24] [332] [357] [367].

Adding objectual quantification to temporal logic allows the formula to capture arithmetical induction, which is the basis for representing full arithmetic [2] [346] [347]. This results in first-order temporal logic being incomplete and the formula is not finitely axiomatisable over models such as the Natural Numbers. Hodkinson et al. [171] presented work on monodic fragments of first-order temporal logic. They showed that monodic fragments have complete axiomatisations and are decidable.

A formula is monodic if the temporal subformulas have only one free variable. For example:

Monodic Formula: $\forall\ x,\ p(x) \rightarrow \bigcirc q(x)$

Not Monodic: $\forall\ x,\ \forall\ y,\ p(x,y) \rightarrow \bigcirc q(x,\ y)$

4.1.4 Summary.

Temporal logic provides a method and notation to place constraints on hypergame models by expressing properties of dynamic systems. It does this by allowing requirements to be stated for the model. Often these requirements have a truth value associated with it, that may vary over time. The truth values can be used to verify the model and ensure the model does not violate the stated constraints over time.

It is well understand that real world conflicts are dynamic systems; the conflict is constantly changing and developing over time. This requires a standard method to express the properties of complex conflicts. Temporal logic provides the necessary structure and notation to express the properties of complex conflicts and can be used with hypergame models.

4.2 Reasoning

"Reasoning is the process of using existing knowledge to draw conclusions, make predictions, or construct explanations." [93] During a conflict is necessary for players to reasoning about the information they have and their beliefs. In a hypergame model the row player is reasoning not only about the strategy to use, but also what strategy the column player may select. In a hypergame model that is repeated over time, reasoning also helps players update their information and beliefs based on what it appears the opponent did previously. There are three methods of reasoning that may be used in hypergame analysis: deductive reasoning where the conclusion is

guaranteed, inductive reasoning where the conclusion is only likely, and abductive reasoning where the conclusion is a best guess.

4.2.1 Deductive Reasoning.

Deductive reasoning starts with a general statement or statements, then reasons to reach a guaranteed logical conclusion. In deductive reasoning general statements are called premises. If the premises are true then the conclusion is also true. For example mathematics is a deductive system:

if x = 2, 0

and if y = 5

then 3x + 2y = 16

Deductive reasoning can also be expressed in ordinary language, which is called a syllogism (such as an If, Then, Else statement from first order logic):

If entropy in a system increases unless energy is expended

And if my bedroom is a system

Then disorder increases in my bedroom until I clean it.

In the syllogism above, the first two statements are premises, and together lead to the third statement, which is the logical conclusion. The conclusion is either sound, true, or unsound, false, which depends on the truth of the original premises. The deductive inference can be either valid or invalid, without regard to the truth or falsity of the original premises. The deductive inference may be valid even when the original premise(s) were false. This means the syllogism results in a false conclusion because one or more premise(s) where false, but the syllogism is still valid since it is logical - the conclusion logically follows from the stated premises.

If premises are sound, then deductive reasoning leads to specific logical conclusions. Deductive reasoning is nonampliative, therefore it cannot increase human knowledge. Conclusions are reached by reducing the premises to tautologies, this allows observations to be made and implications to be expanded, but deductive reasoning cannot be used to predict future or non-observed events.

4.2.2 Inductive Reasoning.

Inductive Reasoning starts with specific observations with limited scope, then reasons to reach a generalized conclusion which is likely, but not guaranteed or certain, based on the observations. Most scientific research uses inductive reasoning where evidence is gathered, patterns are identified and a hypothesis or theory is formed to explain the observations [76].

Conclusions based on inductive reasoning are not logical necessities. The evidence does not guarantee the conclusion, since there is no way to know all the possible evidence has been collected and any new evidence may invalidate the hypothesis or theory. Therefore inductive reasoning is probabilistic, with some probability the conclusion is true, given the premises. For example,

Of the life forms known today, 100% depend on liquid water.

Therefore, any newly discovered life form will probably have some dependence on liquid water.

The above argument can be made every time a life form is discovered and may be correct every time in the past. The truth of the argument in the past does not mean in the future it is impossible to discover life not dependent on water; it only refers to the truth of the argument over the known evidence.

Inductive arguments are not just true or false, since they are not logical necessities. Instead arguments based on induction are cogent (strong) or not cogent

112

(weak). Cogent is when the evidence is complete and relevant, so one is convinced that the conclusion is probably true. Cogent describes how probable the reached conclusion is true.

Inductive reasoning is ampliative, therefore it can increase human knowledge. Since the conclusions are not guaranteed, it can be used to predict future or non-observed events.

4.2.3 Abductive Reasoning.

Abductive reasoning normally begins with an incomplete set of observations, then reasons to reach the likeliest possible conclusion for the set [143]. Charles Sanders Peirce first referred to abduction as "guessing" [279]. He also stated that to abduct a hypothetical conclusion, α, from an surprising observation, β, requires α to be sufficient (or nearly sufficient), but not necessary for β [277]. It is used in daily decision-making processes that make the best use of the information at hand, which is often incomplete. Often abduction results in success that exceeds random luck, Peirce thought that "man have a natural bent in accordance with nature's" which allows man to understand nature through a process called instincts [278]. The following example was offered by Peirce for abduction:

The surprising fact, C, is observed;

But if A were true, C would be a matter of course,

Hence, there is reason to suspect that A is true.

The hypothesis is present in a premise, but its truth not asserted. Then in the conclusion the hypothesis is asserted as rationally likely.

A network administrator trying to recover from a cyber attack is an example of abductive reasoning. Based on a set of symptoms in the network (missing data, unauthorized access, etc.), the administrator must decide what causes explain most of

the symptoms. This is necessary so the problem can be fixed and avoided in the future. Simply stopping the attack and not fixing the cause does not give a healthier network protected from cyber attacks. Cogent inductive reasoning requires fairly complete evidence, while abductive reasoning is defined by a lack of completeness, where either the evidence, explanation, or both are incomplete. When trying to recover from a cyber attack, system logs may be unavailable to report system details as the attacker may have deleted or corrupted the logs to hide the attacker's intentions. Yet, the administer still needs to reach the best causes (thus the best fixes or mitigations) or conclusion based on the incomplete set of network symptoms.

Abductive reasoning can be creative, intuitive, or revolutionary. Peirce stated that abduction in regards to complicated phenomenon lays the "plank of its advance" using plausible, instinctive reason [278]. Abductive reasoning is weak when a fact is used to reason to a potential explanation out of many possible explanations. Strong abductive reasoning is when a fact is used to reason to the best explanation.

Abductive logic is often said to be ignorance-preserving. When knowledge is missing the first response is to acquire new knowledge to reach a solution to the problem and perform some action. The second is to abandon the problem and leave it unsolved while taking no action. The third is abduction, where the problem is unsolved, but the result is still the rationale for some action. This is seen in the area of common law: "When jurors find an accused guilty of the offence with which he has been charges, they do not know whether in fact the offence was committed by him"[368]. While abduction fails to fully answer the ignorance problem, it provides rationale for selecting specific actions [11]. This shows how humans respond to the lack of information; in the case of law, the accused is found guilty and convicted.

The strength of abductive arguments can be increased by following the Surprise Principle and avoiding the Only Game in Town Fallacy [214]. The Surprise Principle

states [336] that for some observation θ, it strongly favors one hypothesis (H_1) over another (H_2) if the following are true:

- If H_1 were true, θ is expected to be true.

- If H_2 were true, θ is expected to be false.

The "Only Game in Town" Fallacy happens when there is one explanation available for a series of surprising events and the reasoning wrongly assumes this only explanation has to be accepted [214]. Lane presents the following example of the "Only Game in Town":

> Hundreds of Americans have reported they have been abducted by space aliens.
>
> The only currently available explanation of this fact is they really have been abducted by space aliens.
>
> Therefore, hundreds of Americans really have been abducted by space aliens.

Lane claims this abductive argument is not strong. Just because there is only one explanation, does not mean the reasoning requires it to be believed. Instead the explanation may be too radical, instead it is possible there is an explanation and it is unknown.

Abduction has been presented as an epistemic process for belief revision [10]

- anomalous (or novel) experience results in surprising phenomenon.

- a state of doubt interrupts the belief habit and triggers abductive reasoning

- goal is to explain the surprising phenomenon and soothe the state of doubt

- doubt is soothed rather than removed because in abduction the hypothesis needs to be tested and proven viable before being converted into a belief

115

According to Woods, "not only does abduction not secure us knowledge, it does not warrant belief" [368]. Abduction only gives a reason to suspect a hypothesis is true, it does not guarantee the hypothesis is true. In the example of common law, all the evidence has already been presented to the jury and there is no way to conduct further testing of hypotheses.

4.3   Belief Revision

Generally beliefs in game theory are represented by a probability distribution over a set of state and Bayes' rule is used to model belief revision. Belief revision is the main building block for two game theoretic solution concepts: perfect Bayesian equilibrium [31] [53] [114] and sequential equilibrium [204].

In perfect Bayesian equilibrium and sequential equilibrium, a player updates his beliefs during a game using Bayes' rule. The information is represented as a set of nodes in the information set. If during the game an information set is reached which has zero prior probability, then the beliefs are formed arbitrarily. From this point forward these new beliefs must be used with Bayes' rule, unless new information is received that is conflicting with the revised beliefs.

Alchourron et al. [7] pioneered belief revision in computer science. Known as AGM theory, it has been widely studied. Gardenfors provides a detailed overview [127]. In AGM theory beliefs are modeled as belief sets, or sets of formulas. Information is represented as a formula and belief revision is modeled as an operation, transforming a belief set into a new belief set that is updated based on the new information. For more information on belief revision with AGM see [255].

According to Bonanno [22], "the tools of modal logic have not been explicitly employed in the analysis of the interaction of belief and information over time." He therefore presents a simple modal logic for iterated belief revision, extending his previous work [54]. For example, based on a patient's symptoms a doctor concludes

the patient is suffering from one of the following: viral infection (V), bacterial Infection (B), medication allergy (M), or food allergy (F). Kripke structures [205] are used to represent the set of states, i.e. $\Omega = $ B, V, M, F. At every time t the doctor's beliefs are represented as a binary relation B_t on Ω. Given some state $\omega \in \Omega$ let $B_t(\omega) = \omega\prime \in \Omega : \omega \ B_t \ \omega\prime$, then $B_t(\omega)$ is the set of states at time period t, that is considered possible when the true state is ω.

From the doctor example above, given the true state is F, the evolution of the doctor's belief may be: $B_0(F) = $ B, V, $B_1(F) = $ V, $B_2(F) = $ M, and $B_3(F) = $ F. Beliefs are updated through the receipt of new information. Information is represented as a sequence of binary relations I_t, for every time t. Therefore, I_t represents an individual learning ϕ at time t. The state ω is then set $\omega \models I_t\phi$ if and only if the following two conditions hold:

- For every $\omega\prime$ such that $\omega I_t \omega\prime, \omega\prime \models \phi$

- For every $\omega\prime \in \Omega$, if $\omega\prime \models \phi$ then $\omega I_t \omega\prime$

These conditions mean that at state ω, $I_t\phi$ is true if the states reachable from ω using the relation I_t coincides with the truth set of ϕ: $I_t(\omega) = \|\phi\|$. The beliefs at time t+1 are the results of the interaction between beliefs at time t and information received at t+1.

4.4 Game Theory and Iteration

There are a few models in game theory that consider time as an element. These include repeated games [230, 231, 242], evolutionary game theory [91, 360], and games with temporal logic. Each of these models handles time in a slightly different matter. Each is discussed, with an overview given focusing on the time element of the model.

117

4.4.1   Repeated Games.

In repeated games, where a base game played for some number of repetitions. Each repetition can be considered as a time epoch where the players must consider how their current action will impact the future actions of the other players in the game. Some of the early work was completed by Ezio Marchi with respect to introducing time into the minimax theorem [230, 231]. There are many models for repeated games in the literature, but in 1989 Mertens, Sorin, and Zamir presented a general model for repeated games [242]. This tuple model includes most other repeated game models as special cases [248].

$$\Gamma^\Gamma \triangleq (N, \Theta, (D_i, S_i, u_i)_{i \epsilon N}, q, p) \tag{4.1}$$

Where N is a nonempty set of players and $i \in N$ is a specific player. The possible states of nature (environment) is represented by Θ, a nonempty set. D_i denotes a set of moves for player i and S_i denotes a set of signals player i may receive, where $D = X_{i \in N} D_i$ and $S = X_{i \in N} S_i$. The initial distribution is given by $q \in \Delta(S \times \Theta)$, the transition function by $p : D \times \Theta \Rightarrow \Delta(Sx\Theta)$, and the payoff function by $u_i : D \times \Theta \Rightarrow \mathbb{R}$.

An example of an infinitely repeated Prisoner's Dilemma is shown in Figure 4.1. Using the repeated game equation (Equation 4.1), the Prisoners' Dilemma can be described by letting N = {1,2}, Θ={0,1}, D_i={f_i, g_i} and $S_i = \Theta \cup (D_1 \text{ x } D_2)$. For each player i $\in$ N and d $\in$ D, let $u_i(d, 1)$ be given by the payoffs listed in Figure 4.1 and let $u_i(d, 0) = 0$. Then let q(1,1,1) = 1, p(d,d,1|d,1) = 0.99, p(0,0,0|d,1) = 0.01, and p(0,0,0|d,0) = 1. Active play continues in state 1 and stops in state 0. The initial distribution q(1,1,1) = 1 means with a probably 1 in the first round, both players will receive a signal "1" and the state of nature will be 1. At each round k+1, if the state of nature is 1, then each player receives a signal with the results of the preceding

round. The probability of continuing is 99% given by p(d,d,1|d,1) = 0.99 and the probability of stopping is 1% given by p(0,0,0|d,1) = 0.01.

Figure 4.1: Prisoners' Dilemma Game Example.

4.4.2 Evolutionary Game Theory.

Another game theoretic model is from evolutionary game theory with replicator dynamics [91, 360]. In evolutionary game theory, the population in the current iteration competes to reproduce or survive to the next population, as shown in Figure 4.2. Each individual in the population will have variation which results in different strategy selection. The game rules test the individual strategies in order to determine the payoff or fitness. This payoff is then used to reproduce or replicate the individual into the next population using replicator dynamics. This produces population (n+1), which takes the place of the previous population and the game is replayed.

Using the Prisoner's Dilemma defined in Figure 4.3, an application of evolutionary game theory (EGT) follows. The game is defined using variables:

119
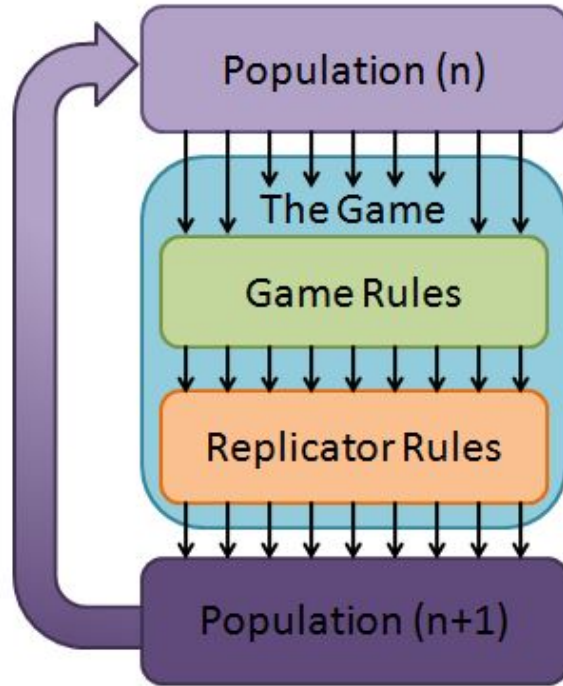
Figure 4.2: Evolutionary Game Theory Model.

- C denotes cooperation

- D denotes defection

- R is the reward attained when all players cooperate

- L is the loser's reward when all players defect

- P is the punishment received by the cooperator when the other player defects

- W is the reward received by the defector when the other player cooperates

- P < L < R < W

In EGT the payoff matrix for each player is $A = \left(\begin{smallmatrix} R & P \\ W & L \end{smallmatrix} \right)$. In the Prisoner's Dilemma, there are two pure strategies, which allows a population with two groups to

Figure 4.3: Evolutionary Prisoner's Dilemma.

be constructed. In the population the frequency of cooperators is x and the frequency of defectors is 1 - x. The strategy frequency vector is then given by $\vec{x} = \left(\frac{x}{1-x}\right)$.

The fitness of the population is then given by $\vec{x}_T A \vec{x}$ and the fitness of the cooperators is then $(A\vec{x})$. If $\vec{x} = \vec{x}(t)$ then the cooperators replicator dynamics is $\dot{x} = x((A\vec{x})_1 - \vec{x}^T A \vec{x})$. The frequency of cooperators converges to 0, leaving only defectors in the population, the evolutionary stable strategy (ESS). This means D, defectors, is a Nash equilibrium and ESS.

4.4.3 Temporal Logic.

Propositional game logic has been used to study the game structure for algebraic properties [274]. This was improved upon by Pauly, who developed Coalition Logic (related to Alternating Temporal Logic) and Game logic (related to Propositional Dynamic Logic and the modal ν-calculus) [276]. Pauly showed relationships between games and software programs using Game Logic, as well as to describe coalitions using Coalition Logic [276]. Other research used a dynamic epistemic language to study the

change of information caused by the player's actions in a game by van Ditmarsch [87], while van Benthem used dynamic logic in order to describe games and the strategies of the players in those games [46].

Alternating temporal logic is applied to a game where players and an environment alternate moves, allowing quantification of the possible outcomes of the game [19]. Goranko has shown the relationship between Pauly's Coalition Logic and alternating temporal logic [140]. These logics focus reasoning on the existence of strategies, while the strategies are not directly considering in reasoning.

van der Hoek et al. provide the foundation for the work of Ramanujam and Simon. They develop logics for strategic reasoning and equilibrium concepts [159, 172]. Ramanujam and Simon [292] present a logic for strategic reasoning and equilibrium concepts using alternating temporal logic. The authors focus on studying strategies by their properties. Strategies can be partial, where they are not completely known as a function instead of atomic (similar to the work of [159, 172]). They use partial strategy specifications, which leads to more generality in reasoning. For example, "(partial) strategy σ ensures the (intermediate) condition α". [292]

4.4.3.1 Game Theoretic Temporal Model.

Ramanujam and Simon present a temporal game model [292] that is divided into discrete sections, as shown in Figure 4.4. The model consists of a game and game arena that provide the foundational elements of the game shown in Figure 4.5. A game defines the structure of the game and is defined as:

$$G \triangleq (N, \Sigma, \Phi, \{\preceq^i\}_{i \in N}) \tag{4.2}$$

Where N is a nonempty finite set of players and $i \in N$ is a specific player. Σ is a nonempty finite set of player actions. A game arena, Φ, defines the rules about game

122

Figure 4.4: Temporal Game Model Presented by Ramanujam and Simon.
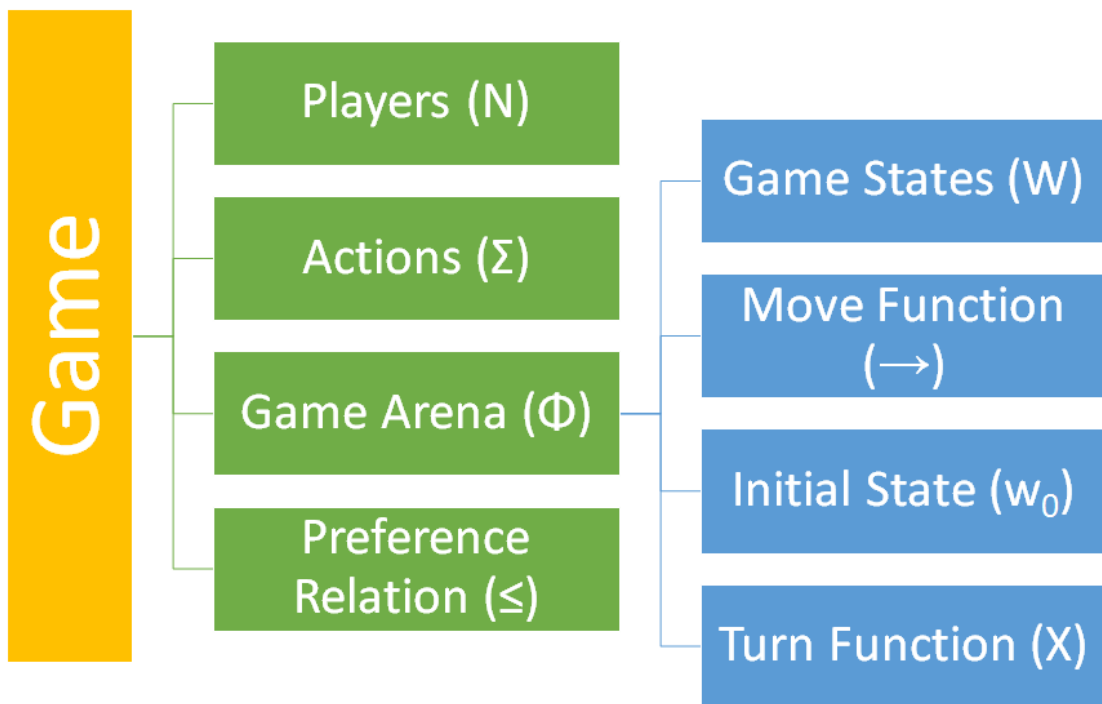
Figure 4.5: Overview of Temporal Game Model.

progression and termination. The preference relation, $\preceq^i$, orders the preferences of player i over the player actions Σ. The preference relation is complete, reflexive, and transitive [292].

A game arena defines the rules about game progression and termination and is defined as:

$$\Phi \triangleq (W, \longrightarrow, w_0, \chi) \tag{4.3}$$

W is a set of game positions or states. The move function, $\longrightarrow$, defines a set of transitions between game positions such that $(Wx\Sigma) \longrightarrow W$. The initial node of the game is given by w_0. The set of successors of $w \in W$ is defined as $\overrightarrow{w} = \{w' \in W \mid w \xrightarrow{a} w' \text{ for some } a \in \Sigma\}$ and the terminal node is defined as $\overrightarrow{w} = \{\emptyset\}$. The function χ assigns every node $w \in W$ the player whose turn it is such that $\chi : W \to N$.

4.4.3.2   Strategy Switching in Temporal Game Model.

Paul et al. propose a method to specify a player's rationale for switching strategies during the course of a game [275]. The focus is on whether players will settle on a strategy without further switching of strategies. The strategy switching notation from [275] is as follows:

- $\sigma_1 \cup \sigma_2$ the player can play according to the strategy σ_1 or the strategy σ_2

- $\sigma_1 \cap \sigma_2$ if σ_1 is defined at a history $t \in \mathcal{T}$ then the player follows σ_1. Else if σ_2 is defined at a history $t \in \mathcal{T}$ then the player follows σ_2. If σ_1 and σ_2 are defined at history t, then the moves(actions) specified by σ_1 and σ_2 at history t are the same. σ_1 and σ_2 are said to be compatible and the $\cap$ operator is used to denote compatible pairs of strategies.

- $\sigma_1 \frown \sigma_2$ the player plays the strategy σ_1 and then after some time (based on history), switches strategies and plays σ_2. The position in time of the strategy switch is unknown in advance and not fixed.

- $(\sigma_1 + \sigma_2)$ allows the player to choose either strategy σ_1 or σ_2 at every point of the game

- $\psi?\sigma$ based on the history of game play, the player tests if the property ψ holds and if it holds the player according to σ

4.4.3.3   Branching Time Frames.

Surowik [344] uses temporal logic of branching time first presented by A. N. Prior[287] which leads to an indeterministic temporal logic. Modification of the structure of time removes determinism, making the logic indeterministic [344]. Temporal logic of branching time is used to model the dynamic interactions between players in a temporal context, given the notion of process. The author models extensive games in temporal logic by adding a notion of agent and a notion of prediction to the semantics [344].

Surowik starts with the notion of branching-time frames presented by Bonanno [55]. A branching-time frame with agents (BTA-frame) is a tuple:

BTA-frame = $\langle$T, $\prec$, $R_{ii \in N}\rangle$

- T âĂŞ a set of nodes

- $\prec$ - a binary relation on T (precedence relation) satisfying

  A1) antisymmetry: if $t_1 \prec t_2$, then $t_2 \not\prec t_1$

  A2) transitivity: if $t_1 \prec t_2$ and $t_2 \prec t_3$, then $t_1 \prec t_3$

  A3) left linearity: if $t_1 \prec t_3$ and $t_2 \prec t_3$, then $t_1 = t_2$ or $t_1 \prec t_2$ or $t_2 \prec t_1$

- N = $\{1,...,n\}$ a finite set of agents

- R_i is a binary relation on T (R_i a subrelation of $\prec$) - if $t_1 R_i t_2$, then $t_1 \prec t_2$ for any $I \in N$

The properties of antisymmetry (A1), transitivity (A2), and left linearity (A3) contain the definition of branching time, with the left linearity property limits the possible frames to frames where the past is unique and there are alternatives in the future at any node[344]. The binary relation R_i allows an agent, denoted i, to make a decision which leads to a node where the agent does not have any available actions.

The notion of prediction is added to a BTA-frame by defining the prediction binary relation as $\prec_p$ on T, such that [344]:

B1) if $t_1 \prec_p t_2$, then $t_1 \prec t_2$ ($\prec_p$ is a subrelation of $\prec$)

B2) $t_1 \prec_p t_2$ and $t_2 \prec_p t_3$, then $t_1 \prec_p t_3$

B3) if $t \prec t_1$ for some t_1, then $t \prec_p t_2$, for some t_2

B4) if $t_1 \prec t_2$, $t_2 \prec t_3$ and $t_1 \prec_p t_3$, then $t_1 \prec_p t_2$ and $t_2 \prec_p t_3$

The BTA-frame is then used to build a game model based on the extensive form with perfect information. For perfect information games the following sentences are introduced:

S1) $\mu_i = q$, where i $\in$ N and q $\in$ Q i.e. the player's payoff

S2) $q_1 \leq q_2$, i.e. the payoff q_1 is less than or equal to q_2

Given this a BTA frame for a perfect information game can be created, denoted $\mathcal{T}$. The game model $\mathcal{M}$ is based on $\mathcal{T}$ with an added valuation function V : s $\longrightarrow 2^T$ with the following conditions [344]:

- if the sentence is of the form of S1, then V(S1) = T

- if the sentence is of the form of S2, then V(S2) = {t ∈ L(T): $\mu(t) = q$}

The game model can then be used to determine the truth of formulas that describe properties of games. For example, the author models the game theoretic concept of backward induction with the context of temporal logic branching time.

Samuel Reid [297] presents a similar temporal logic model for game theory. A branching-time frame is used, but a history is added in order to be able to determine if the formulas are satisfied. A history h ∈ H in T is a maximal linearly ordered subset of T, where H is the set of all histories in T [297]. Articulated histories are used to split histories into past and future time points. Reid's definition of an articulated history and time point instant is:

- *An articulated history of a time point $t \in T$ is a pair $(h_p(t)$, $(h_f(t))$, where $(h_p(t) = \{t^{'} — t^{'} ¡ t\}$ is the set of all past time points of t in all histories containing t and $h_f(t) = \{t^{'} — t^{'} ¡ t\}$ is a set of future time points of t which determines a unique history $h = h_p(t) \bigcup \{t\} \bigcup h_f(t)$*

- *A set of time points $\{t_1, …, t_n\}$ belongs to an instant $I \subseteq T$ if $t_i \nprec t_j$ and $—h_p(t_i)— = —h_p(t_j)$, ∀ i,j*

These definitions then allow a branching-time frame, $\mathcal{T}$, be a game, G. If Ψ is a formula that denotes a tie between players, Reid gives the following definition of a game:

- *In an n-player game G, the Instant I_i is player k's turn if $k \equiv I$ mod n. Furthermore, player k wins the game at a turn t in a history h if $|h_f(t)| = 0$ and $t \in I_i$ with $k \equiv i − 1$ mod n and $G_{h,t} \models \rightarrow \Psi$*

4.4.3.4   Other Approaches.

Agotnes et al. discuss irrevocable strategies under Alternating-time Temporal Logic (ATL) [4]. In ATL in order to achieve a goal, an agent chooses a strategy and then follows the strategy without considering what other agents do, which leads to the goal always being true. Although in ATL, the strategies are revocable, where the agent is not restricted by previous choice of strategies resulting in the state where the goal is evaluated. Irrevocable strategies are often assumed in game theory [4]. The authors discuss variants of ATL where the strategies are irrevocable. The aim is to further multi-agent logic by aligning them with strategies in game theory. The two variants are IATL (memory-less irrevocable strategies) and MIATL (memory-based irrevocable strategies). In IATL the model is updated directly by fixing the choices of every agent in a given state as defined by the agent's chosen strategy. In MIATL, it is not possible to update the model directly; instead the model must be represented as a tree-like structure where the branches that are not compatible with the current strategy are eliminated.

Anticipation games where developed in order to handle concurrent interactions among players [65]. Anticipation games are based on game theory [269] using Timed Alternating-time Temporal Logic (TATL) to model concurrent interactions by executing time-based rules [164]. Using timed-based rules the model can incorporate an element of surprise where one player receives an advantage over the other [9]. In the Attacker-Defender environment, surprise may happen by the attacker using a zero-day exploit or the defender patching a known vulnerability before the attack can exploit it [358]. Anticipation games have been extended to determine the cost and time needed to eliminate an attack [63, 64].

4.5 Summary

In this chapter, temporal logic is explored as a foundation for adding time to the hypergame model, which has currently not been researched. Reasoning is discussed as a way to formalize the decision making in the hypergame analysis of the players. Different types of reasoning can be used by the players at different points of the game. For example, in hypergames with strategic surprise a player would use abductive reasoning in order to try to reasoning about missing information.

The foundation provided in this chapter is built upon in the following chapters as hypergame theory is extended with temporal logic and abductive reasoning is used in the game model. The next chapter provides an overview of the problem domain and discusses how temporal logic is used to extend hypergame theory. Chapter three also provides the problem domain for the application of abductive logic to hypergame theory.

V. Hypergame Models

This chapter develops the hypergame models in mathematical notation for clear understanding. There are two distinct hypergame models presented in the background literature. The first was presented by Bennett [34, 39], providing the foundation of hypergame theory. The second was presented by Vane in his doctoral dissertation work [356], building on the work by Bennett. Vane's model was later extended by Gibson [135].

5.1 Bennett's Model

The original hypergame representation proposed by Bennett in his seminal papers on hypergame theory [34, 39], is based on using ordered sets of outcomes. The representation consist of two different, but related, game theoretic models. Using Bennett's original hypergame definitions and the notation presented by Fraser and Hipel [106], a hypergame is:

Definition 9. *Bennett Hypergame is given by $H \triangleq \{G_1, G_2, ..., G_n\}$ where G_i is the perceived game by player i.*

A game, G, is defined as:

Definition 10. *A game is given by $G \triangleq \{N, \theta, (\Sigma, \mu)_{i \in N}, \leq\}$*

Where N is the set of players in the hypergame from the view of the player under consideration. The set of possible game states, θ, is given for the perceived game. For each player in the perceived game, there is a set of strategies/actions and a payoff function μ, such that $\mu : \Sigma \times \theta \Rightarrow \mathbb{R}$. Finally, a preference relation, $\leq$, over μ is used to order the payoffs of the game.

5.2 Vane's Model

The second hypergame model is described by Vane in the literature. Vane expands on Bennett's hypergame theory by developing Hypergame Normal Form (HNF). HNF added hyperstategies to the hypergame model, in addition to the Nash equilibrium solution. Each hyperstategy represents a solution to the hypergame where the player in question may be able to obtain an outcome better than the Nash equilibrium.

Vane also adds belief contexts to the model in order to represent the belief that the opponent chooses a particular strategy. Risk is incorporated into the model through the fear of being outguessed. Vane restricts Bennett's hypergame model by only focusing on a single player's view during game play. While this appears to reduce the information available in the model, a given player would have their game perception and can incorporate what their opponent does into the belief contexts.

For a hypergame in HNF, the model is:

Definition 11. *Hypergame Normal Form (HNF) is given by $HNF_{row} \triangleq \{G_{row}\}$ where G_{row} refers to the row player and is the primary point-of-view during game play.*

The HNF model only contains a single game, since it only models the perception of the row player and does not consider the perceptions of other players. A game, G, is defined as:

Definition 12. *A game in HNF is given by $G \triangleq \{N, \theta, \beta, (\Sigma_{Full}, \mu)_{i \in N}, \leq, \gamma\}$*

Where N is the set of players in the hypergame from the view of the player under consideration. The set of possible game states, θ, is given for the perceived game. The belief of the primary point-of-view player is represented by β. For each player in the perceived game, there is a set of strategies/actions Σ_{Full} and a payoff function μ, such that $\mu : \Sigma_{Full} \times \theta \times \beta \Rightarrow \mathbb{R}$. The set of strategies/actions Σ_{Full} is made up of two

types of strategies: normal and hidden. Therefore $\Sigma_{Full} = \{\Sigma_{Normal}, \Sigma_{Hidden}\}$, where Σ_{Hidden} represent strategies that are not believed to be considered by the player, but are available as valid in the game. Finally, a preference relation, $\leq$, over μ is used to order the payoffs of the game. The fear of being outguessed, γ, is a percentage such that $0 \leq \gamma \leq 1$.

5.3 Extension of Vane's Model

Gibson extends Vane's HNF model to use variable payoff/utility functions [135]. Hypergames initially have only been represented using an ordered set of outcomes which represents preference as Bennett has done. Gibson updates the hypergame model with the ability to have variables in payoff functions. This update requires variable definitions, such as payoff function variables, initial variable values, variable update function, and variable constraints.

Bennett's original hypergame model can also be updated to use variable payoff functions. While the HNF model provides more realistic modeling, this model is provided for completeness. For a perceived game, denoted G, the updated model for hypergames with variable payoff functions is:

Definition 13. *Variable Payoff Functions* *are represented in a game by* $G \triangleq \{N, \theta, \beta, (\Sigma_{Full}, \mu)_{i \in N}, \leq, \mathcal{V}, \gamma\}$

Most of the parameters are the same between Vane's model and Gibson's model. Gibson's model adds, $\mathcal{V}$, which are payoff function variable definitions from the view of the primary point-of-view player. The payoff function variable definitions can be defined as:

Definition 14. *Payoff function variable definitions* *are given by* $\mathcal{V} \triangleq \{\omega, \lambda, \psi, \sigma\}$

where ω are the payoff function variables. The initial variable values at the start of game play is represented by λ. The variable update function(s) are represented by ψ and the constraints placed on the variables during updating are defined as σ. The payoff function μ, must now use $\mathcal{V}$, such that $\mu : \Sigma_{Full} \times \theta \times \beta \times \mathcal{V} \Rightarrow \mathbb{R}$.

5.4 Extension of Theorems

Sasaki [310] presents theorems (Theorems 10, 12, 11 and Lemma 13) that all focus on a simple hypergame. Simple hypergames are hypergame where the only difference between the perceived games is in the outcomes. This means the current theorems are not directly applicable to the hypergame model presented by Vane [356].

In Vane's model (as well as numerous examples from Bennett's work [34]) there is overlap of the perceived games between the players. The lack of overlap often occurs because of additional strategies (called hidden strategies in Vane's model). A base game similar to Sasaki's Theorem 11 is defined as:

Finding 1. *Let $H = (G^p, G^q)$ be a hypergame with $G_p = (N, \Sigma_p, u_p)$ and $G_q = (N, \Sigma_q, u_q)$ where $p, q \in N$. A normal form game $G = (N, \Sigma, u)$ is called the base game of H iff $u = u_p$, $u = u_q$, and $\Sigma = \Sigma_p \cap \Sigma_q \neq$. Let the base game (BG) of hypergame H be denoted by BG_H.*

A simple conclusion from the previous Finding, is that subgames always have the same base game under Vane's model.

Finding 2. *Subgames always have the same base game*

The game that remains after considering the BG can also be defined. This part of the game is important because this is where misperception and deception can happen. The difference game of a hypergame is:

Finding 3. *Let $H = (G^p, G^q)$ be a hypergame with $G_p = (N, \Sigma_p, u_p)$ and $G_q = (N, \Sigma_q, u_q)$ where $p, q \in N$. A normal form game $G = (N, \Sigma, u)$ is called the difference*

game of H iff $u = u_p \cup u_q \setminus u_p \cap u_q$, $\Sigma = \Sigma_p \cup \Sigma_q \setminus \Sigma_p \cap \Sigma_q$, and $u \neq$ or $\Sigma \neq$. Let the difference game of hypergame H be denoted by Δ_H.

The hyper Nash equilibrium solution for a simple hypergame can also be extended to include differences in strategies as well as outcomes in gmaes. The definition is the same as Sasaki [310] but inclues the additional constraint that the strategy profile must be in both perceived games. Lemma 13 is still valid according to the previous finding.

Finding 4. *Let $H = (G^p, G^q)$ be a hypergame with $G_p = (N, \Sigma_p, u_p)$ and $G_q = (N, \Sigma_q, u_q)$. Then $a^* \in \Sigma_p \cap \Sigma_q$ is called a stable hyper Nash (SHN) equilibrium iff $a^* \in N(G_p)$ and $a^* \in N(G_q)$ where N(G) represents the Nash equilibriums for game G.*

5.5   Summary

This chapter defines the hypergame models found in the literature using concise mathematical notion. The two major models from Bennett and Vane, as well as the extended by Gibson are covered. Theorems presented in other literature are extended to better represent Vane's model in this chapter. The mathematical notion for the models serves as the foundation of integrating hypergames and temporal constructs for a new framework. In the next chapter the hypergame models are integrated with temporal logic to from a temporal hypergame framework.

VI. Temporal Hypergame Model

Given the background on hypergame models (by Bennett and Vane), as well as game theoretic temporal aspects, a temporal hypergame model can be used to reason about conflicts, specifically the decisions and the misperceptions of the participating players. This chapter aims to unify the two approaches, creating a temporal hypergame model. A base temporal model is presented, representing Bennett's original hypergame model [34].

This base temporal model is the foundation refined using the enhancements provided by Vane and Gibson. Vane's Hypergame Normal Form (HNF) adds four enhancements to the base temporal model including beliefs, fear of being outguessed, hyperstrategies, and hidden strategies (to support the restriction of row's point-of-view) [356]. Gibson enhances Hypergame Normal Form (HNF) by adding support for variable payoff functions [135]. Using this temporal hypergame framework, the concepts of trust, misperception, and deception are constructed.

6.1   Base Hypergame Model

The temporal hypergame model builds on the game theoretic temporal model by adding perceptions. A high level overview of the temporal hypergame model is shown in Figure 6.1. Each hypergame is composed of multiple games, with each game having a game arena.

In a first level hypergame, each player may have an unique view of the game, or conflict. This results in a hypergame being formulated as a set of games, one from each player. A superscript "T" is used to denote temporal models. For n players, a temporal hypergame is defined as a set:
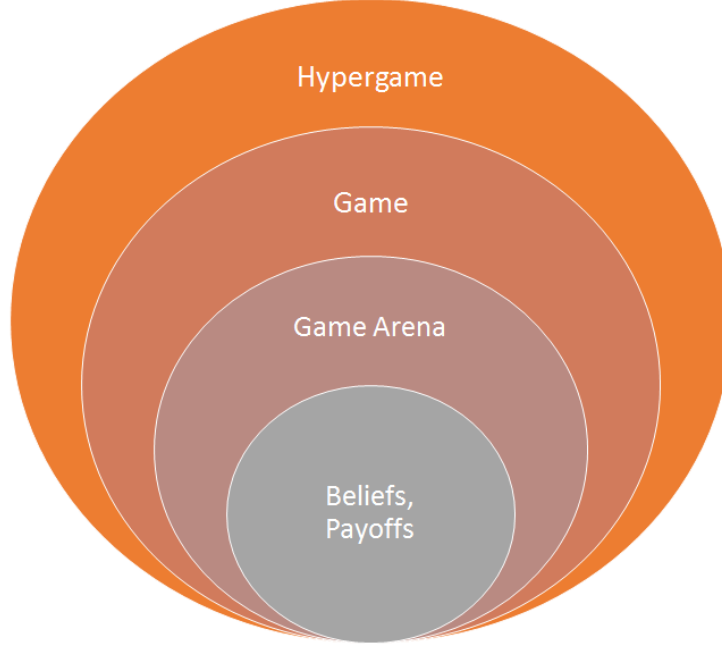
Figure 6.1: Overview of Temporal Hypergame.

Definition 15. *A **Temporal Hypergame** is given by $H^T \triangleq \{G_1^T, G_2^T, ..., G_n^T\}$ where G_i^T represents the perceived game by the ith player. The perceived game specifies the winning conditions, which specify the game outcomes.*

Definition 16. *A **Perceived temporal game** in a hypergame is given by $G^T \triangleq \{N, \Sigma, \Phi^T, \leq_{i \in N}^i\}$ where N is a non-empty set of players (at least two). Σ represents the non-empty finite set of player actions. A game arena, Φ^T, defines the rules about game progression and termination. A preference relation, $\leq^i$, is given for each player and represents the perceived preferences of the player instead of actual preferences.*

A game arena, Φ^T, is a finite graph defining the rules about game progression and termination. A game arena is defined as:

Definition 17. *A **Game Arena** in a temporal game is given by $\Phi^T \triangleq \{W, \rightarrow, w_0, \chi\}$ where the set of game positions, w, represents the outcomes and possible states as the*

136

game progresses. The move function, $\to$, is responsible for transitioning the game. The initial node, or position of the game, is represented by w_0. $\chi : W \to N$ is the function that assigns every node $w \in W$ the player whose turn it is at the node.

In this model the hypergame provides perceptions and subgames, the game provides structure, and the game arena defines rules for progression and termination, as shown in Figure 6.2. Another view of the model is through the learning/decision making process defined in Figure 3.1. The hypergame models the player's perceptions, the game models the structure of the learning/decision making process, and the game arena provides the rules for orienting and observing, as well as for feedback and learning.
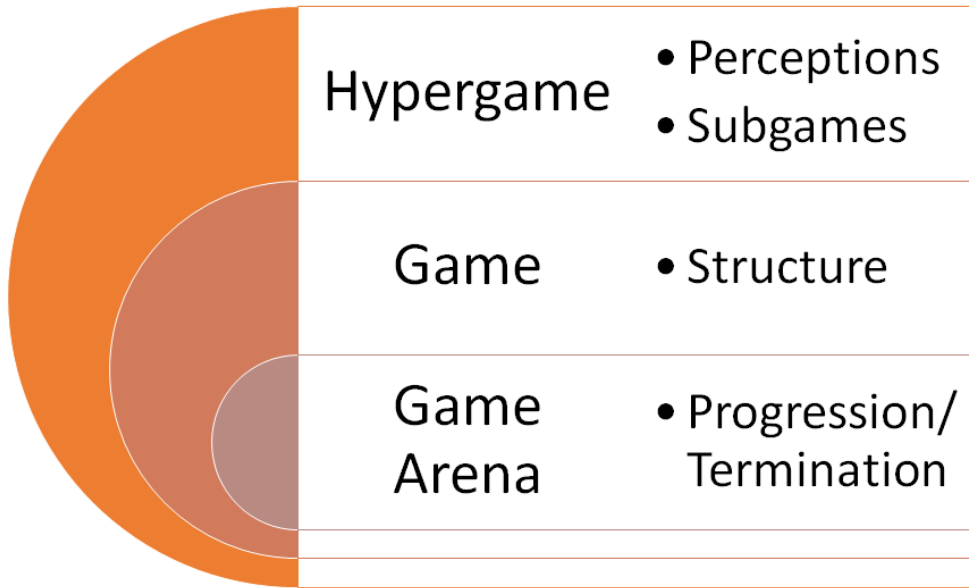
Figure 6.2: Temporal Hypergame Structure.

6.2 HNF Refinements

The Hypergame Normal Form proposed by Vane [356] offered a few refinements to the original hypergame model. These refinements include the row's belief of the what the column player does, the fear of being outguessed, Row Mixed Strategy (RMS) and Column Mixed Strategy (CMS), and the focus on row's point of view. Gibson further refined the Hypergame Normal Form (HNF) model to use variable update functions. These additions need to be integrated into the temporal model. An outline of how to integrate beliefs, fear-of-being-outguessed, hyperstrategies, and row's point-of-view is given in symbolic form.

6.2.1 Beliefs.

Beliefs are incorporated into the Hypergame Normal Form (HNF) model in two different ways. The first is through the use of Column Mixed Strategy (CMS), which represents the row player's beliefs about the percentages the other player is using to select a strategy. These can be known prior or calculated by finding the Nash Equilibrium of subgames. The second is through belief contexts, which represents the row player's beliefs the other player is using the particular strategy. The game arena Φ^T. is updated as follows to reflect the row player's beliefs:

Definition 18. *A Game Arena with Beliefs* is given by $\Phi^T \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \chi)$

where β is the belief contexts over the game, and β_0 is the initial set of beliefs for the row player about the strategies the other player will choose to play. $\mathcal{C}$ is the Column Mixed Strategy (CMS) over the game, while $\mathcal{C}_0$ represents the initial set of Column Mixed Strategy (CMS) values for the row player. The $\hookrightarrow$ is the update function for the row player's beliefs, where $W_{current} \times \beta \times \leq_{i \in N}^i \hookrightarrow \beta_{new}$ and $W_{current} \times \mathcal{C} \times \leq_{i \in N}^i \hookrightarrow \mathcal{C}_{new}$. The remaining symbols retain the previous meaning.

6.2.2 Fear-of-being-outguessed.

The HNF model uses the fear-of-being-outguessed to represent a player's willingness to accept risk by deviating from the Nash equilibrium and introduce strategic surprise into the game. At the heart of the HNF model, the fear-of-being-outguessed, measures uncertainty the row player has about the game. This is just one measure of uncertainty, and other measures can be included into the model in a similar way if they lead to better strategy selection and ultimately to better payoffs/outcomes.

The fear-of-being-outguessed can be static for the entire game or it can be dynamically update based on game play and the row player's beliefs. It can cause oscillation between strategies (including mixed) and may not lead to equilibrium solution. This is especially important with dynamically updating the variable. These effects can be reduced by updating the fear-of-being-outguessed based on maximizing the column player's expected utility rather than minimizing the row player's expected utility.

The game arena Φ^T is updated as follows to reflect the fear-of-being-outguessed:

Definition 19. *A Game Arena with the Fear-of-being-Outguessed is given by* $\Phi^T \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \Upsilon, \twoheadrightarrow, \gamma, \chi)$

where Υ is the fear-of-being-outguessed over the game, and γ is the initial fear-of-being-outguessed. The fear-of-being-outguessed has the following constraint: $0 \leq \Upsilon, \gamma \leq 1$. The $\twoheadrightarrow$ is the update function for the row player's fear-of-being-outguessed, where $\Upsilon_{current} \times Wcurrent \twoheadrightarrow \Upsilon_{new}$. This also has an affect on the move function in the game arena by influencing the possible move states: $W \times \Upsilon \xrightarrow{a} w'$.

6.2.3 Hyperstrategies.

Given the progression of hypergames to date (Bennett, Vane, Gibson) and considering the element of strategic surprise, it is necessary to represent the ability of the players to switch strategies. This is especially true when looking at hyperstategies

such as Row Mixed Strategy (RMS). Row Mixed Strategy (RMS) are based on what the row player believes (or perceives) the column player is going to play in the game. Since only one strategy applies to the whole game (represented by the Nash equilibrium), the row player would be free to switch to a hyperstrategy if he believed the other player was playing a subgame. Players can start with a set of potential strategies (hyperstrategies) and switch between each at will, which could be event based, time based, etc. Paul, *et al.* [275] provide a syntax for representing players' rationale for switching strategies which is used to represent strategy switching in the hypergame model. A strategy set with switching is defined as:

Definition 20. *Strategy set with switching* *in a hypergame is given by* $\Pi_{new} :\triangleq$ $\sigma \in \Sigma_i | \sigma_1 \cup \sigma_2 | \sigma_1 \cap \sigma_2 | \sigma_1 \frown \sigma_2 | \sigma_1 * \sigma_2 | \psi ? \sigma$

The expanded meaning of the strategy switching syntax is as follows:

- $\sigma_1 \cup \sigma_2$ the player can play according to the strategy σ_1 or the strategy σ_2

- $\sigma_1 \cap \sigma_2$ if σ_1 is defined at a history t $\in \mathcal{T}$ then the player follows σ_1. Else if σ_2 is defined at a history t $\in \mathcal{T}$ then the player follows σ_2. If σ_1 and σ_2 are defined at history t, then the moves(actions) specified by σ_1 and σ_2 at history t are the same. σ_1 and σ_2 are said to be compatible and the $\cap$ operator is used to denote compatible pairs of strategies.

- $\sigma_1 \frown \sigma_2$ the player plays the strategy σ_1 and then after some time (based on history), switches strategies and plays σ_2. The position in time of the strategy switch is unknown in advance and not fixed.

- $(\sigma_1 * \sigma_2)$ allows the player to choose either strategy σ_1 or σ_2 at every point of the game

- $\psi?\sigma$ based on the history of game play, the player tests if the property ψ holds and if it holds the player according to σ

Strategy switching does not guarantee the game results in a stable solution. At any point of time during a game when strategy switching occurs, the game can move to a different equilibrium. It is only possible to move to a different equilibrium and is not guaranteed.

6.2.4 Row's Point-of-View.

In HNF, Vane restricted the model to only the row's point-of-view. This removes the additional perceived games by other players. In reality the perceived games of other players may not be known, except in cases of historical analysis (which is the primary use for the Bennett version of the hypergame model). In Vane's model the importance is placed on the idea of subgames, or games that vary in a same way, from the row's master perceived game.

Definition 21. *HNF model with only row's point-of-view is given by* $HNF_{row} \triangleq \{G_{row}\}$

where G_{row} refers to the row player and is the primary point-of-view during game play. The HNF model only contains a single game (a restriction of Bennett's original model), since it only models the perception of the row player and does not consider the perceptions of other players.

Definition 22. *A Game is defeined as* $G^T \triangleq \{N, (\Sigma_{Full})_{i \in N}, \Phi^T, \leq^i_{i \in N}\}$

The set of strategies/actions Σ_{Full} is made up of two types of strategies: normal and hidden. Therefore $\Sigma_{Full} = \{\Sigma_{Normal}, \Sigma_{Hidden}\}$, where Σ_{Hidden} represent strategies that are not believed to be considered by the player, but are available as valid in the game.

141

6.2.5 Variable Payoff Refinements.

Another refinement to the HNF model is from Gibson [135] who integrated variable payoff functions into the model. The variable payoff functions are defined by a set of variables, initial values, an update function, and a set of constraints. The game arena Φ^T is updated as follows to reflect the variable payoff functions:

Definition 23. *Temporal Game Arena with Beliefs and fear-of-being-outguessed is given by $\Phi^T \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \Upsilon, \twoheadrightarrow, \gamma, \mathcal{V}, \chi)$*

The addition to the game arena is the $\mathcal{V}$ representing variable payoff functions.

Definition 24. *Variable Payoff Functions is defined as $\mathcal{V} \triangleq \{\omega, \omega_{init}, \psi, \delta\}$ where ω are the payoff function variables. The initial variable values are given by ω_{init}. The variable update function is ψ subject to the constraints defined by δ.*

6.3 The Temporal Model

The temporal work of Ramanugam and Simon [292] serves as the foundation of the hypergame temporal model. The temporal model is extended to incorporate hypergame theory, including Bennett's and Vane's models. A superscript "T" is used to denote temporal models. An overview of the variables used in the temporal hypergame framework are shown in Figure 6.3. The model is presented with the hypergame refinements for completeness. For n players, the temporal hypergame is defined as:

Definition 25. *Temporal Hypergame is denoted as $H^T \triangleq \{G_1^T, G_2^T, ..., G_n^T\}$ for $n \in N$ where G_i^T represents the perceived game by the ith player. The perceived game specifies the winning conditions, which specify the game outcomes.*

Definition 26. *Perceived Temporal Game is denoted by $G^T \triangleq (N, \Sigma_{Full}, \Phi^T, \{\preceq^i\}_{i \in N})$*
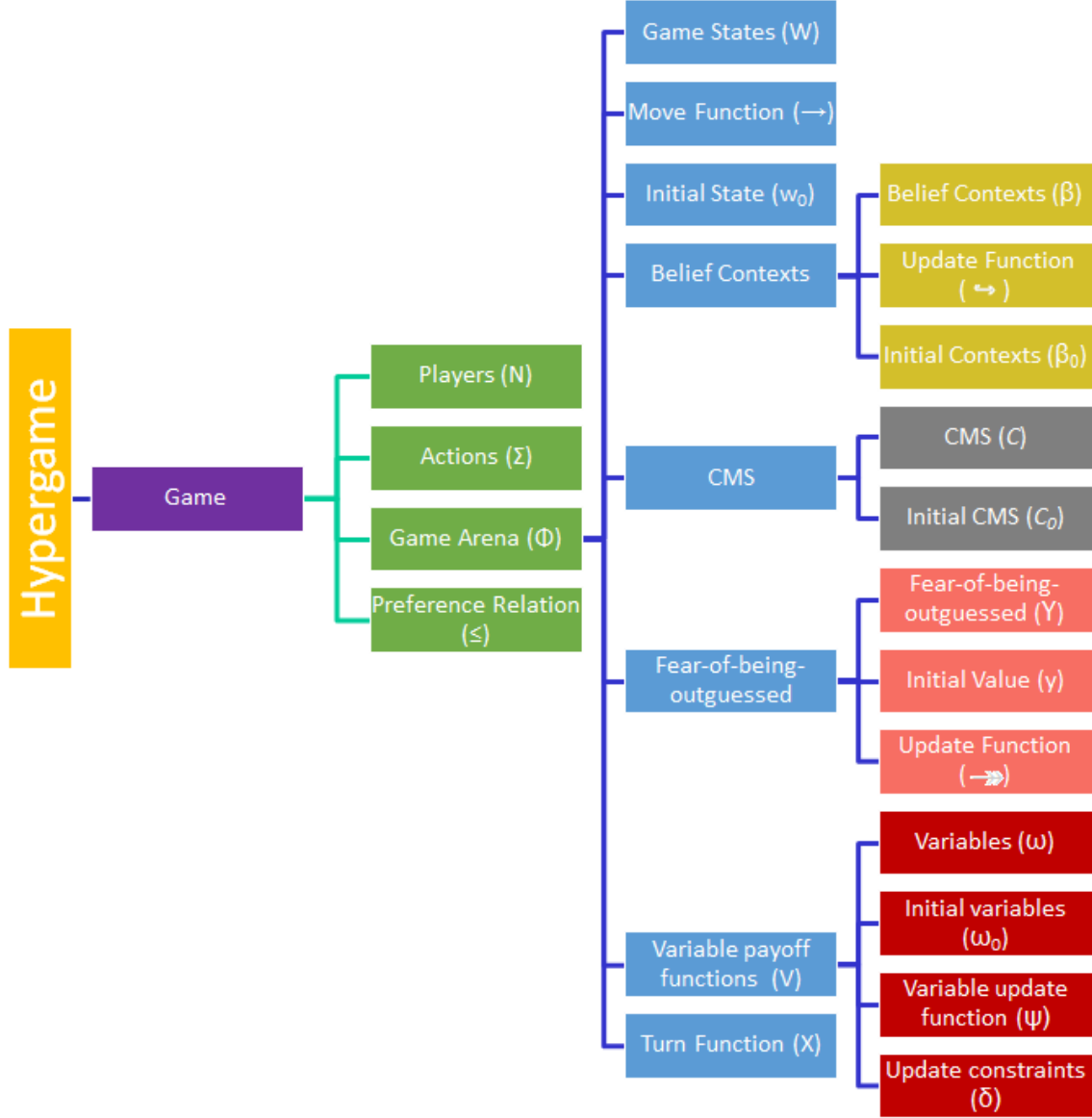
Figure 6.3: Temporal Hypergame Framework Variables.

Where N is a nonempty finite set of players and $i \in N$ is a specific player. Σ is a nonempty finite set of player actions. A game arena, Φ, defines the rules about game progression and termination. The preference relation, $\preceq^i$, orders the preferences of player i over the player actions $\Sigma Full$. The set of strategies/actions Σ_{Full} is made up of two types of strategies: normal and hidden. Therefore $\Sigma_{Full} = \{\Sigma_{Normal}, \Sigma_{Hidden}\}$,

143

where Σ_{Hidden} represent strategies that are not believed to be considered by the player, but are available as valid in the game. The preference relation is complete, reflexive, and transitive.

A game arena, Φ, is a finite graph defining the rules about game progression and termination:

Definition 27. *A **Temporal Game Arena** is denoted by $\Phi^T \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \Upsilon, \twoheadrightarrow, \gamma, \mathcal{V}, \chi)$ where W is a set of game positions or states. The move function, $\longrightarrow$, defines a set of transitions between game positions such that $(Wx\Sigma) \longrightarrow W$. The initial node of the game is given by w_0. The set of successors of $w \in W$ is defined as $\overrightarrow{w} = \{w' \in W \mid w \xrightarrow{a} w'$ for some $a \in \Sigma_{Full}\}$ and the terminal node is defined as $\overrightarrow{w} = \{\emptyset\}$.*

The belief contexts, β, over the game and β_0 is the initial set of beliefs for i^{th} row player about the strategies the other player chooses to play. $\mathcal{C}$ is the Column Mixed Strategy (CMS) over the game, while $\mathcal{C}_0$ represents the initial set of Column Mixed Strategy (CMS) values for the row player. The $\hookrightarrow$ is the update function for the row player's beliefs, where $W_{current} \times \beta \times \leq_{i \in N}^i \hookrightarrow \beta_{new}$ and $W_{current} \times \mathcal{C} \times \leq_{i \in N}^i \hookrightarrow \mathcal{C}_{new}$.

The fear-of-being-outguessed, Υ, over the game represents the risk of selected a hyperstrategy and γ is the initial fear-of-being-outguessed. Υ is subject to the following constraint: $0 \leq \Upsilon, \gamma \leq 1$. The $\twoheadrightarrow$ is the update function for the row player's fear-of-being-outguessed, where $\Upsilon_{current} \times Wcurrent \twoheadrightarrow \Upsilon_{new}$. This also has an affect on the move function in the game arena by influencing the possible move states: $W \times \Upsilon \xrightarrow{a} w'$.

The addition to the game arena is the $\mathcal{V}$ representing variable payoff functions. The function χ assigns every node $w \in W$ the player whose turn it is such that $\chi : W \to N$. If $\chi(w) = k$ and w is not a terminal node then player k must choose an action at w.

When $\chi(w_0) = $ k, player k owns the game position w_0 and must choose an action 'a' available at w_0. The new game position is then w' where $w_0 \xrightarrow{a} w'$. A play (denoted p) in Φ is a sequence of moves, or infinite path, p $= w_0 \xrightarrow{a_1} w_1 \xrightarrow{a_2} \ldots \xrightarrow{a_k} w_k$ where w_k is a terminal node or the path is infinite: p $= w_0 \xrightarrow{a_1} w_1 \xrightarrow{a_2} \ldots$ where $\forall$ i : $w_i \xrightarrow{a_i} w_{i+1}$.

Definition 28. *Variable Payoff Function is denoted by* $\mathcal{V} \triangleq \{\omega, \omega_{init}, \psi, \delta\}$ *where* ω *are the payoff function variables. The initial variable values are given by* ω_{init}. *The variable update function is* ψ *subject to the constraints defined by* δ.

An extensive form game tree $\mathcal{T}$ can be associated with a game arena, Φ.

Definition 29. *Extensive Form Game Tree is denoted by* $\mathcal{T} \triangleq (S, \Rightarrow, s_0, \lambda)$

The countably infinite tree, $(S, \Rightarrow)$, is rooted at s_0 with edges from Σ_{Full}. The function λ maps the nodes of the tree to game positions - $\lambda : S \rightarrow W$. Where λ has the following properties:

- $\lambda(s_0) = w_0$

- $\forall s, s' \in S$, if s $\xrightarrow{a} s'$ then $\lambda(s) \xrightarrow{a} \lambda(s')$

- if $\lambda(s) = $ w and w $\xrightarrow{a} w'$ there exists $s' \in$ S s.t. s $\xrightarrow{a} s'$ and $\lambda(s') = w'$

With the extensive form game tree $\mathcal{T}$ and some node s, a restriction of $\mathcal{T}$ to s, denoted $\mathcal{T}_f$, is the unique path from the root s_0 to s, the tree root.

If $\mathcal{T}$ is the extensive form game tree and s_1 is a node in $\mathcal{T}$. At s_1 a strategy for player 1 is defined as $\nu = (S_\nu^1, S_\nu^2, \Rightarrow_\nu, s_1)$. ν is a subtree of $\mathcal{T}_{s_1}$ which represents the unique path from s_0 to s_1 in $\mathcal{T}$ satisfying:

- $s_1 \in S_\nu^1$, where $\chi(\lambda(s_1)) = 1$

- $\forall$ s $\in \mathcal{T}_G$ rooted at s_1; if s $\in S_\nu^i$ then for some $a_i \in \Sigma$, $\forall$ s$'$ such that s $\xrightarrow{a_i} s'$, then s $\xRightarrow{a_i}_\nu s'$

6.3.1 Hypergame Strategies.

A strategy for a given player i is defined by a function $\nu^i : S^i \rightarrow \Sigma_{Full}$, which specifies the moves for the player at every game position. If $\mathcal{T}$ is the extensive form game tree and s_1 is a node in $\mathcal{T}$. At s_1 a strategy for player 1 is defined as $\nu = (S_\nu^1, S_\nu^2, \Rightarrow_\nu, s_1)$. The strategy ν can be viewed as a subtree of $\mathcal{T}$. This means that for each node assigned to player i, there is a unique outgoing edge and for nodes belonging to other players every available move is included [134]. The strategy tree is defined as $\mathcal{T}_\nu = (S_\nu, \Rightarrow_\nu, s_0, \hat{\lambda}_\nu)$ associated with ν is a least subtree of $\mathcal{T}$ which represents a unique path in $\mathcal{T}$ satisfying:

- $s_0 \in S_\nu$

- For any s $\in S_\nu$

    - if $\hat{\lambda}$(s) = i, there exists a unique $s' \in S_\nu$ and action a s.t. s $\overset{a_i}{\Rightarrow}$ s'

    - if $\hat{\lambda}$(s) $\neq$ i, $\forall$ s' s.t. s $\overset{a_i}{\Rightarrow}$ s', then s $\overset{a_i}{\Rightarrow}_\nu$ s'

$\Omega_i(\mathcal{T})$ denotes the set of player i's strategies in game G with extensive form game tree $\mathcal{T}$. A play p : $s_0 a_0 s_1$ in the game, is consistent with ν if $\forall$ j $\geq$ 0, $s_j \in S^i$, then $\nu(s_j) = a_j$. A strategy profile ⟨ν_i, ν_{i+1}, ..., ν_N ⟩results in a play $p_{\nu_i}^{\nu_{i+1},...,\nu_N}$ that is unique in G.

6.3.2 Hypergame Partial Strategies.

Strategy specifications can be partial, in which a player is allowed to assume an opponent plays α whenever p holds without considering or knowing the conditions that would cause the opponent to pick another move b in the opponent's strategy. Partial strategies lead to more generality in reasoning with the temporal logic.

A partial strategy for a given player i is defined by a function $\sigma^i : S^i \rightharpoonup \Sigma_{Full}$, which specifies the moves for the player at some, but not every, game position. The domain of the partial function σ^i is denoted by $\mathcal{D}_{\sigma^i}$. The partial strategy σ can be

viewed as a subtree of $\mathcal{T}$. This means that for some nodes that belong to player i, there is an unique outgoing edge and for player i's other nodes (as well as the nodes belonging to the other players) every move is included [134].

The partial strategy tree is defined as $\mathcal{T}_\sigma = (S_\sigma, \Rightarrow_\sigma, s_0, \hat{\lambda}_\sigma)$ associated with σ is a least subtree of $\mathcal{T}$ which represents a unique path in $\mathcal{T}$ satisfying $s_0 \in S_\nu$. For any s $\in S_\nu$, if $\hat{\lambda}$(s) = i and s $\in \mathcal{D}_{\sigma^i}$, there exists a unique $s' \in S_\nu$ and action a s.t. s $\overset{a_i}{\Rightarrow} s'$. Otherwise if $(\hat{\lambda}$(s) = i and s $\notin \mathcal{D}_{\sigma^i})$ or $\hat{\lambda}$(s) $\neq$ i, $\forall s'$ s.t. s $\overset{a_i}{\Rightarrow} s'$, then s $\overset{a_i}{\Rightarrow}_\nu s'$.

Partial strategies can be represented by a set of total strategies, and any total strategy can be represented as a partial strategy, where the set of total strategies is s singleton set. For the partial strategy σ and the partial strategy tree $\mathcal{T}_\sigma$ the set of total strategy trees $\hat{\mathcal{T}}_\sigma$ is defined as $\mathcal{T} = (S, \Rightarrow, s_0, \hat{\lambda}) \in \hat{\mathcal{T}}_\sigma$ if and only if the following are true:

- if s $\in \mathcal{T}$ then $\forall s' \in \vec{s}$, $s' \in \mathcal{T}$ implies $s' \in \mathcal{T}_\sigma$

- if $\hat{\lambda}$(s) = i, there exists a unique $s' \in S_\sigma$ and action a such that s $\overset{a}{\Rightarrow}_\sigma s'$

6.3.3   The Logic - Strategy Specification.

The logic for reasoning about composite strategies is divided into two parts: strategy specification and game formulas. Atomic strategy formulas specify the conditions a player tests before making a move. Connectives are then used to construct composite strategy specifications from the atomic strategy formulas. Game formulas represent the model logic description of the game area. It specifies the results of a player executing a strategy, choosing a move a, and ensuring an outcome at some intermediate time α.

The following preliminaries are useful before describing the logic and its semantics. For some countable set $\mathcal{X}$, let the past combinations of the members of X be denoted by Past($\mathcal{X}$):

147

$$\psi \in \text{Past}(\mathcal{X}) := \text{x} \in \mathcal{X} \mid \neg\psi \mid \psi_1 \vee \psi_2 \mid \diamond\psi$$

These formulas obtain meaning when applied over a finite sequence. For example, given any sequence $\xi = t_0 t_1 \ldots t_m$, $V : \{t_0, \ldots, t_m\} \to 2^{\mathcal{X}}$, and n such that $0 \leq \text{n} \leq$ m, ξ, n $\models \psi$ (the truth of the formula $\psi \in \text{Past}(\mathcal{X})$ at k) as:

- ξ, n $\models$ p iff p $\in V(t_k)$

- ξ, n $\models \neg\psi$ iff ξ, n $\not\models \psi$

- ξ, n $\models \psi_1 \vee \psi_2$ iff ξ, n $\models \psi_1$ or ξ, n $\models \psi_2$

- ξ, n $\models \diamond\psi$ iff $\exists$ j : $0 \leq$ j $\leq$ n such that ξ, j $\models \psi$

Let $P^i = \{p_0^i, p_1^i, \ldots\}$ represents a set of proposition symbols where $\tau_i \in P^i$, for I $\in$ N. Let P $= \bigcup_{i \in N} P^i \cup \{\text{leaf}\}$. τ_i represents which player's turn it is to move at a given game position. A terminal node is specified by leaf. The finite set of player's moves, $\sigma = \{a_1, a_2, \ldots, a_m\}$, parameterizes the logic when game areas over σ are considered.

Definition 30. *Let* $Strat^i(P^i)$, *for* i $\in$ N *be the strategy specification set given by:* $Strat^i(P^i) := [\psi \mapsto a_k]^i \mid \sigma_1 + \sigma_2 \mid \sigma_1 \cdot \sigma_2 \mid \pi_\iota \Rightarrow \sigma$

where $\pi_\iota \in Strat^\iota (\bigcap_{i \in N} P^i)$ (the other player i's strategy specification), $\psi \in$ Past(P^i) and $a_k \in \sigma$.

Definition 31. *Let* $Switch^i$, *for* i $\in$ N *be the strategy specification set with strategy switching given by:* $Switch^i := \sigma \in \Sigma_i | \sigma_1 \cup \sigma_2 | \sigma_1 \cap \sigma_2 | \sigma_1 \frown \sigma_2 | \sigma_1 * \sigma_2 | \psi?\sigma$

For a partial strategy denoted by σ, the strategy switching operators have the following expanded meaning:

- $\sigma_1 \cup \sigma_2$ the player can play according to the strategy σ_1 or the strategy σ_2

- $\sigma_1 \cap \sigma_2$ if σ_1 is defined at a history $t \in \mathcal{T}$ then the player follows σ_1. Else if σ_2 is defined at a history $t \in \mathcal{T}$ then the player follows σ_2. If σ_1 and σ_2 are defined at history t, then the moves(actions) specified by σ_1 and σ_2 at history t are the same. σ_1 and σ_2 are said to be compatible and the $\cap$ operator is used to denote compatible pairs of strategies.

- $\sigma_1 \frown \sigma_2$ the player plays the strategy σ_1 and then after some time (based on history), switches strategies and plays σ_2. The position in time of the strategy switch is unknown in advance and not fixed.

- $(\sigma_1 * \sigma_2)$ allows the player to choose either strategy σ_1 or σ_2 at every point of the game

- $\psi?\sigma$ based on the history of game play, the player tests if the property ψ holds and if it holds the player according to σ

These constructs allow the properties of strategies to be specified, resulting in using the combination of constructs to describe game play. The meaning of $[\psi \mapsto a_k]^i$ for $p \in P^i$ is that player i chooses move "a" when it is that player's turn and p holds. At positions of the game where it is player i's move and p does not hold, any enabled move is allowed to be instead. The construct $\sigma_1 + \sigma_2$ means that the strategy of player i conforms to the specification σ_1 or σ_2, while the construct $\sigma_1 \cdot \sigma_2$ means the strategy conforms to specification σ_1 and σ_2. $\pi_\iota \Rightarrow \sigma$ says that player i sticks to the specification σ if the history of play reveals all moves made by ι conforms to π_ι. This captures the game theoretic view that the actions of players are in responses to the opponent(s) moves and the play is forced to make a move on the history game play without knowing the opponent(s) complete strategy.

These constructs are formalized for a game tree $\mathcal{T}$ and a node $s \in \mathcal{T}$ and the strategy specification $\sigma \in Strat^i(P^i)$. The least subtree of $\mathcal{T}_f$ is defined as: $\mathcal{T}_f \upharpoonright \sigma$

$= (S_\sigma, \Rightarrow_\sigma, s_0)$ which contains the unique path from s_0 to s, denoted $p^s_{s_0}$. $\mathcal{T}_f \upharpoonright \sigma$ is closed under the following conditions:

- $\forall\, s' \in S_\sigma$ such that s $\Rightarrow^*_\sigma s'$

    - s' is an i node: $s' \overset{a}{\Rightarrow} s''$ and a $\in \sigma(s') \Leftrightarrow s' \overset{a}{\Rightarrow}_\sigma s''$

    - s' is an i/node: $s' \overset{a}{\Rightarrow} s'' \Leftrightarrow s' \overset{a}{\Rightarrow}_\sigma s''$

For the game tree $\mathcal{T}$ and a node, s $\in \mathcal{T}$, let the unique path from s_0 to s be $p^s_{s_0}$: $s_0 \overset{a}{\Rightarrow} s_1 \cdots \overset{a_m}{\Rightarrow} s_m =$ s. Given the strategy specification $\sigma \in Strat^i(P^i)$ and node s $\in \sqcup$, the definition of σ(s) is:

- $[\psi \mapsto a]^i(s) = \begin{cases} \{a\} & \text{if s} \in W^i \text{ and } p^s_{s_0}, \text{m} \models \psi \\ \sigma & \text{otherwise} \end{cases}$

- $(\sigma_1 + \sigma_2)(s) = \sigma_1(s) \cup \sigma_2(s)$

- $(\sigma_1 \cdot \sigma_2)(s) = \sigma_1(s) \cap \sigma_2(s)$

- $(\pi \Rightarrow \sigma)(s) = \begin{cases} \sigma(s) & \text{if } \forall\, j : 0 \le j < m,\, a_j \in \pi(s_j) \\ \sigma & \text{otherwise} \end{cases}$

Path $p^{s'}_s$: s $= s_1 \overset{a_1}{\Rightarrow} s_2 \cdots \overset{a_{m-1}}{\Longrightarrow} s_m = s' \in \mathcal{T}$ is said to conform to σ if $\forall\, j : 1 \le j < m,\, a_j \in \sigma(s_j)$. Play is said to conform to σ when the path leads to proper play.

The following null is used to represent an empty specification and is defined as:

Definition 32. *An empty specification* *is denoted by* $null^i \triangleq [\top \mapsto a_1] + \cdots + [\top \mapsto a_m]$

Any strategy for player i conforms to $null^i$, which is useful for asserting a strategy exists while the property of the strategy are irrelevant.

6.3.4 Strategy Specification Syntax.

The logic syntax is given by P and Π:

Definition 33. ***The strategy logic syntax*** *is given by* $\Pi :\triangleq p \in P \mid \neg\alpha \mid \alpha_1 \vee \alpha_2$
$\mid < a^+ > \alpha \mid < a^- > \alpha \mid \diamond\alpha \mid (\sigma)_i : c \mid \sigma \rightsquigarrow_i \beta$

where $c \in \Sigma$, $\sigma \in Strat^i(P^i)$, $\beta \in \text{Past}(P^i)$. Let $\diamond\alpha = \neg\boxminus\neg\alpha$, $< N^+ > \alpha = \bigvee_{\alpha \in \Sigma} < a^+ > \alpha$, $[N]\alpha = \neg < N > \neg\alpha$, $< P > \alpha = \bigvee_{\alpha \in \Sigma} < a^- > \alpha$, and $[P] = \neg < P > \neg\alpha$. Other standard connectives are used: $\wedge$ (conjunction), $\supset$ (if then), and $[a]\alpha$.

The formula $(\sigma)_i$: c means that at any position in the game the player i's strategy specification σ suggests the move c can be played at the given position. The formula $\sigma \rightsquigarrow_i \beta$ means that from the current position in the game, there exists a way to follow player i's strategy σ where β is ensured to be the outcome.

6.3.5 Semantics.

Extensive form game trees and a valuation function serve as the model for the logic. This results in a model $\mathcal{M} \triangleq (\mathcal{T}, \mathcal{V})$ where game tree $\mathcal{T} \triangleq (S^1, S^2, \rightarrow, s_0)$ defined from before, and the valuation function $\mathcal{V} : S \rightarrow 2^P$ such that the following is true:

- For i $\in$ N, $\tau_i \in \mathcal{V}(s)$ iff s $\in S'$

- leaf $\in \mathcal{V}(s)$ iff moves(s) $= \emptyset$

where for a node s, moves(s) $= \{a | s \overset{a}{\Rightarrow} s'\}$.

The truth of formula $\alpha \in \Pi$ in the model $\mathcal{M}$ and at position s, denoted $\mathcal{M}, s \models \alpha$, is defined on the structure of α. Let $p_{s_0}^s : s_0 \overset{a_0}{\Rightarrow} s_1 \cdots \overset{a_{m-1}}{\Rightarrow} s_m = s$.

- $\mathcal{M}, s \models$ p iff p $\in \mathcal{V}(s)$

- $\mathcal{M}, s \models \neg\alpha$ iff $\mathcal{M}, s \not\models \alpha$

- $\mathcal{M}, s \models \alpha_1 \vee \alpha_2$ iff $\mathcal{M}, s \models \alpha_1$ or $\mathcal{M}, s \models \alpha_2$

- $\mathcal{M}, s \models <a+> \alpha$ iff $\exists s' \in W$ such that $s \xrightarrow{a} s'$ and $\mathcal{M}, s' \models \alpha$

- $\mathcal{M}, s \models <a-> \alpha$ iff $m > 0$, $a = a_{m-1}$ and $\mathcal{M}, s_{m-1} \models \alpha$

- $\mathcal{M}, s \models \diamond\!\!\!\cdot\, \alpha$ iff $\exists\, j : 0 \leq j \leq m$ such that $\mathcal{M}, s_j \models \alpha$

- $\mathcal{M}, s \models (\sigma)_i : c$ iff $c \in \sigma(s)$

- $\mathcal{M}, s \models \sigma \rightsquigarrow_i \beta$ iff $\forall s' in \mathcal{T}_s \upharpoonright \sigma$, such that $s \Rightarrow^* s'$ then $\mathcal{M}, s' \models \beta \wedge (\tau_i \supset enabled_\sigma)$.

where $enabled_\sigma \equiv \bigvee_{a \in \Sigma}(<a> True \wedge (\sigma)_i : a)$.

Satisfiability and validity can be defined for a function, α. α is said to be satisfiable if and only if there exists a model $\mathcal{M}$, and there exists s such that $\mathcal{M}, s \models \alpha$. α is said to be valid if and only if $\mathcal{M}, s \models \alpha, \forall \mathcal{M}, \forall s$.

6.4 Trust, Misperception, and Deception

At the foundation of human decision making is trust. When people have trust in others they are dealing with, they assume more risk or cooperate in the face of incomplete or imperfect information. This makes trust a central concept in order to formulate the key idea of deception in a hypergame model.

The concepts of trust and distrust can be found discussed throughout literature (see [232] or [148]). Using the temporal hypergame model, the concept of trust is defined within the constrains of the model. Given a formal definition of trust, a definition of distrust is then constructed. From these two concepts a formal definition of deception is given.

Recall the temporal hypergame framework presented in this chapter. Given a temporal hypergame model $H^T \triangleq \{G^T\}$ (Definition 15), a perceived temporal game $G^T \triangleq \{N, \Sigma, \Phi^T, \leq^i_{i \in N}\}$ (Definition 16), and the extensive form game tree $\mathcal{T}$ associated

with the game arena Φ^T. For two strategies σ_1 and σ_2 in G^T with N = $\{1, 2\}$ with the extensive form game tree $\mathcal{T} \triangleq (S, \Rightarrow, s_0, \lambda)$ the concept of trust, distrust, misperception, and deception is defined.

6.4.1 Trust.

In modeling conflicts and decision-making problems, trust is an important concept. Trust can determine how players respond to perceived devaitions in the game, or which strategies the players prefer. Trust is defined in the temporal hypergame framework by:

Finding 5. *Weak Trust - Player 1 has weak trust in Player 2 if at a vertex $s' \in S_{\sigma_1}$ $\exists\, s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

Finding 6. *Strong Trust - Player 1 has strong trust in Player 2 if $\forall\, s' \in S_{\sigma_1}\, \exists\, s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

The distinction between weak and strong trust is important during decision making or human interactions. Weak trust (Definition 5) covers the case of when there is only one vertex in the extensive form game tree where the strategy leads to expected play/outcome. Strong trust (Definition 6) covers the case of when all vertices in the extensive form game tree lead to the expected play/outcome for the selected strategy.

While it is important to trust other players in a game, it is also important to know which players to distrust. Distrust is defined using the definitions of trust in Definition 5 and Definition 6.

Finding 7. *Weak Distrust - Player 1 has weak distrust in Player 2 if at a vertex $s' \in S_{\sigma_1}\, \nexists\, s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

Finding 8. *Strong Distrust - Player 1 has strong distrust in Player 2 if $\forall\, s' \in S_{\sigma_1}\, \nexists\, s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

Mistrust occurs when a player has distrust (as defined in either Definition 7 or Definition 7) in the other player. Formally this is defined as:

Finding 9. *Mistrust* *- Player 1 mistrust Player 2 if $\exists\ s' \in S_{\sigma_1}$ where Player 1 distrusts Player 2.*

6.4.2 Misperception.

The concept of misperception is closely related to trust, distrust, and mistrust. When a player misperceives something, then they recognize it incorrectly in some way. Misperception is an important concept in hypergames, and is necessary to gain an advantage. A formal definition of misperception over the temporal hypergame framework is:

Finding 10. *Misperception* *- Player 2 misperceives the strategy of Player 1 if there is at least one vertex $s' \in S_{\sigma_1} \exists\ s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$ so that when a new strategy σ_* which is equal to σ_1 except that $\nexists\ s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s^*$ and $(\sigma_1,\ \sigma_2) \leq^1 (\sigma_*,\ \sigma_2)$*

6.4.3 Deception.

If a player intentionally takes advantage of a misperception in a game, then the player is said to deceive the other player(s). Formally deception is defined over the temporal hypergame model:

Finding 11. *Deception* *- Player 1 (p) deceives Player 2 (q) if for a hypergame H, the following is true:*

- *Player 2 trusts Player 1, according to Finding 5 or Finding 6.*

- *Player 2 misperceives the strategy of Player 1, according to Finding 10.*

- *If there exists a strategy pair $(\sigma_p,\ \sigma_q)$, $\sigma_p \in \Sigma_p$ and $\sigma_q \in \Sigma_q$ where $(\sigma_p,\ \sigma_q) \in N(G_T)$ and $(\sigma_p,\ \sigma_q) \notin SHN(G_T)$ and $(\sigma_{p^*},\ \sigma_q) \in SHN(G_T)$*

The first condition clearly states that player 2 must trust player 1. The second condition states that player 1 must have a strategy that is only different in one way from the expected strategy. The final condition states that the strategy pair with the deception strategy must be a Nash equilibrium in the hypergame, but not a stable hyper Nash equilibrium and that the strategy for the deceived player must be a stabel hyper Nash equilibrium with the strategy that is misperceived.

6.5   Additional Findings on the Temporal Hypergame

Given the Theorem 4 stating that there exists a SPNE in a game with finite repetitions and the Theorem 10 where there is a hyper Nash equilibrium in every finite game with mixed strategies. Then it can be concluded:

Finding 12. *In every finite temporal hypergame with mixed strategies, there is at least one SPNE (which may be in the base game).*

If a stage game has a Nash equilibrium then the repeated game has a SPNE (in infinitely repeated games) according to Theorem 8 and Lemma 9.

Finding 13. *If a temporal hypergame H at some time x has a Nash equilibrium in the base game (i.e. stable hyper Nash equilibrium), then the temporal hypergame has a SPNE.*

Given Finding 12 and Finding 13 then the following is true:

Finding 14. *In both the infinite and finite temporal hypergame with mixed strategies, there is at least one SPNE.*

According to the one-shot deviation principle (Theorem 6) states that a strategy is a SPNE if and only if there is no profitable one-shot deviation. The implication of this applied to hypergames is:

155

Finding 15. *In a hypergame H, a strategy is a SPNE in the base game iff there is no profitable one-shot deviation. A one-shot deviation would produce a strategy in the difference game of hypergame H.*

6.6 Summary

This chapter develops and discusses a temporal hypergame framework, representing a new frontier with hypergames. The temporal hypergame framework started with the original hypergame model from Bennett and is enhanced with the models provided by Vane and Gibson. The concepts of trust, misperception, and deception are given using the constructs of the temporal hypergame framework in symbolic form. This framework is applied in the next chapter to the Prisoner's Dilemma and Chicken (classical game theoretic examples), as well as a more complex SCADA network (attacker/defender) game.

VII. Temporal Hypergame Cyber Physical System Applications

This chapter presents the application of the temporal hypergame framework to different hypothetical conflicts. First, an overview on verification and validation for models is given. Next, a discussion on why the examples where chosen is presented. The temporal hypergame framework is applied to a classical game theoretic game, an iterated hypergame, and three cyber physical system examples.

The first application involves the classical game theoretic example of the Prisoner's Dilemma. The framework is applied to' the Prisoner's Dilemma in order to show the framework is valid with classical games. The detailed mathematical description is included in Appendix C. The example focuses on the repeated (or iterated) Prisoner's Dilemma where the same game is played over and over by the players. This example shows the ability of the temporal hypergame framework to model the Prisoner's Dilemma and demonstrate the reasoning concept of the backwards induction.

In real world players often reason using iterated hypergames. Each hypergame in the iterated series, has unstable outcomes that are intended to give the player an advantage in a later iteration [199]. The hypergame is the same game in each iteration, as is done in the repeated Prisoner's Dilemma. This is shown in second application, when the temporal hypergame framework is applied to an iterated hypergame.

The temporal hypergame framework builds on this concept of iteration, but allows the model to change with time which is more representative of real-world conflicts. The temporal hypergame framework does not require the hypergame to be the same game as in the iterative hypergame, instead aspects of the hypergame can vary from time epoch to time epoch. Aspects that can vary include available actions

to a player, the way the outcomes are calculated, and the way the player chooses a strategy.

In this chapter, the complete temporal hypergame framework is applied to one game theoretic and two additional examples with hypergame properties, in order to better demonstrate the temporal utility of the framework. By definition of hypergames, the temporal hypergame framework provides a way to find the strategies equal to or better than the Nash equilibrium. The examples show that the framework results in an appropriate result for the hypergame structure.

7.1 Verification and Validation

Models (like the temporal hypergame framework) are built in order to gain a better understanding of complex systems [185]. A model is a logical structure used to suggest the progression and conclusion of the represented system over time. The system complexity, which may have infinite parts and interactions, can make it difficult to characterize and understand the complete system [209]. The objective of the model is to reduce the system complexity by identifying aspects of the system that allow a satisfactory, but not always perfect understanding. While models are not right or wrong, some may provide more insight into a system than others. Various models can be compared on the basis of how the model helps in system understanding.

Because a model is based on a few chosen aspects, to fully exercise the model a system is required to have representative measures for the chosen aspects. If the system to be modeled does not demonstrate the aspects of interest or only a partial set of the aspects, then the model may not produce realistic results or may exercise part of the model, but still give realistic results. *The utility of a model is based how accurately the measures extracted from the model collate to the measures extracted from the represented system.*

158

There are two ways to determine the utility of the model with respect to the system. The first way is to determine if the model implements the assumptions correctly through model verification [151]. The second, is to determine whether the assumptions are reasonable for the represented system through model validation[151]. While one way does imply the other, both are commonly used together. The goal of the verification and validation process is to show the model represents the system being modeled at a fidelity needed for decision-making [186]. Verification and validation also show the model is credible at an acceptable level for usage by the decision-maker.

There are two ways to conduct verification and validation. The first, is through mathematical proofs that give high fidelity results that the assumptions hold in the model for given cases. The second, through justification by example gives a lower fidelity result that the assumptions may hold in the model for the specific case and are representative of the specific modeled system [318]. *This chapter uses justification by example to validate the utility of the hypergame temporal framework.*

When using justification by example, it is not possible to test 100% as testing every conceivable combination of inputs (game models) is not possible. The structural properties of the games chosen as inputs to the temporal hypergame framework determine which aspects of the framework are tested. While not every input can be tested, a subset of the structural properties can provide a satisfactory level of confidence that the framework behaves as expected.

A chosen specific game may not have the structural properties that exercise all aspects of the framework. Normally a game has a subset of properties that can exercise some partial aspects of the framework. This means more than one game may be needed to exercise more aspects of the framework to determine if it has correct behavior.

159

In the case of the Prisoner's Dilemma (discussed in Appendix C), the structural properties of the game do not allow for detailed hypergame analysis which is a testing objective. With the Prisoner's Dilemma, the game is symmetric and does not have differences in perception required for hypergame analysis. For improved hypergame analysis, a game should have asymmetric structural properties such as a player with an unknown action/strategy or misperception about the value or ordering of the game outcomes. The simple cyber physical system example and the extended cyber physical system example are designed to exercise the additional structural properties for improved hypergame analysis.

7.2 Iterated vs. Temporal Hypergames

In game theory, a repeated game (also known as an iterated game [366]) consists of some number of repetitions of a base game [254]. The base game (also known as a stage game) remains exactly the same through the repetitions (or time) [294]. This same definition and concept can be applied to hypergames, where the repeated hypergame does not change over time or from iteration to iteration.

How does an iterated hypergame relate to a temporal hypergame? The idea is that any iterated hypergame can be represented using a temporal hypergame (shown in Figure 7.1). In doing this, the temporal hypergame should produce the exact same result (outcome) as the iterated hypergame, without additional information or knowledge being incorporated into the temporal model.

The temporal hypergame cannot produce a different result from the iterated hypergame, by definition of an iterated and temporal hypergame. If different results were produced then the two methods would not be related as shown in Figure 7.2. This view is not supported in this research, as this research is based on the relationship in Figure 7.1.
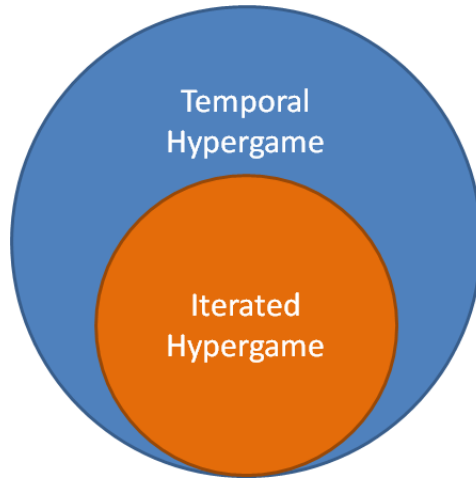
Figure 7.1: Iterated and Temporal Hypergame Relationship.
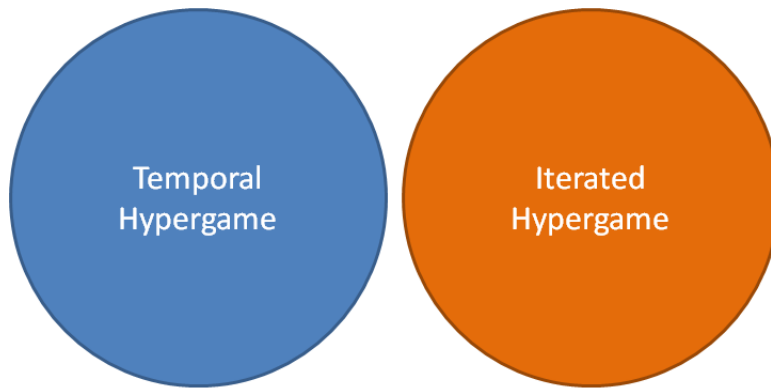
Figure 7.2: Alternative Hypergame Method Relationship.

This is demonstrated in Appendix C, where the iterated (repeated) Prisoner's Dilemma results in the same outcome when using the iterated or temporal methods. The outcome is the same under both methods, and backward induction applies in both cases. Not every temporal hypergame can be represented as an iterated hypergame.

7.3 Model Application Steps

There are six basic steps to applying the temporal hypergame framework presented in Section 6.3 of Chapter 6, as shown in Figure 7.3. The first step, initial game definition, is where the game structure is defined in terms of players and actions. The next step, identification of states and transitions, is where the game progression rules are developed. In the game mapping step, the game is mapped to the extensive form game tree. Path structuring develops the game model and allows the simplification of reasoning about paths in the game tree. With the basic structure, progression, and transition of the game, the next step builds upon the rules and game constraints in order to define player strategies. Finally, using the constructs developed in the previous steps the temporal hypergame framework is used to analyze the game.

Figure 7.3: Temporal Hypergame Framework Process Overview.

7.4 Iterated Attacker-Defender Hypergame Application

Gibson [135] presents a hypergame model of an attacker-defender network game. The game is based on the original game theoretic model presented by Chen and Leneutre [72]. A detailed overview is given in Appendix B. The game presented by Gibson is used here with modified outcomes. The outcomes are modified to allow manual definition of the game while allowing uncluttered symbolic definitions.

The game is shown in Figure 7.4 in normal form. The defender has four actions: not defend, defend, provide ruse, and shutdown. The attacker has three actions: not

162

attack, attack, and zero-day exploit. The attacker-defender game is also shown in Figure 7.5 as a game tree in extensive form.

Attacker

Defender		Not Attack	Attack	Zero-Day Exploit
	Not Defend	(0, 0)	(-3, 3)	(-3, 3)
	Defend	(-2, 0)	(5, 3)	(5, 3)
	Provide Ruse	(-2, 0)	(6, 3)	(5, 3)
	Shutdown	(-2, 0)	(4, -1)	(5, -1)

Figure 7.4: Attacker-defender Game Normal Form.
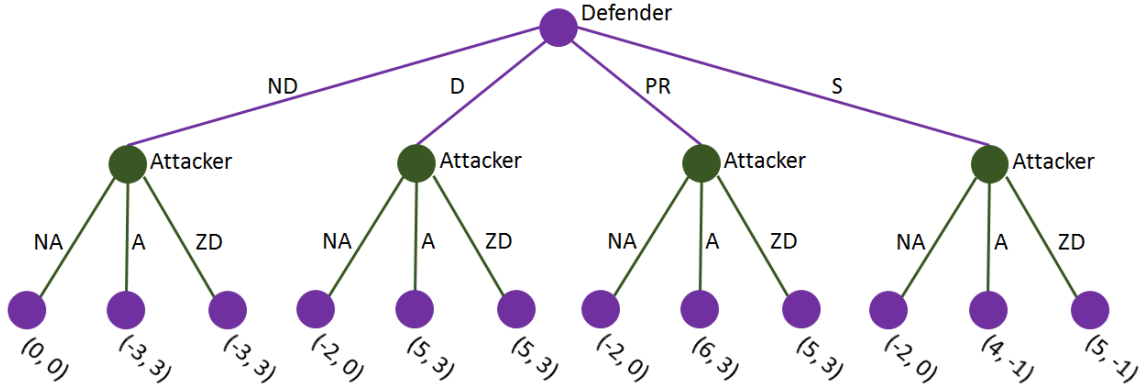
Figure 7.5: Attacker-defender Game Extensive Form.

7.4.1 Initial Game Definition.

To apply Step 1: Initial Game Definition of the temporal hypergame framework to the information from the network model, it is necessary to create a temporal hypergame model. Let the temporal hypergame be:

$$H_{net}^T \triangleq \{G_D^T\}$$

163

The perceived game G_{net}^T:

$$G_{net}^T \triangleq (N, \Sigma_{Full}, \Phi^T, \{\preceq^i\}_{i \in N})$$

Where N = {Attacker (A), Defender (D)} is the set of players. Σ_{Full} is the finite set of player actions such that $\Sigma_{Full} = \Sigma_A \cup \Sigma_D$, $\Sigma_A = \{NotAttack(NA), Attack(AT),$ $Zero-DayExploit(ZD)\}$, and $\Sigma_D = \{NotDefend(ND), Defend(DE), ProvideRuse(PR),$ $andShutdown(S)\}$. In this model the action provide ruse and shutdown is known to the defender, but not the attacker.

The preference ordering function for the i_{th} player $\preceq^i = u_i(x, y) \le u_i(x', y)$ for i $\in$ N. This means for attacker $\preceq^A$ indicates the outcome (x,y) is preferred to outcome (x', y) if x $\le x'$ $\forall (x, y) \in W$. For defender $\preceq^D$ indicates the outcome (x,y) is preferred to outcome (x, y') if y $\le y'$ $\forall (x, y) \in W$. Notice the Nash equilibrium concept is defined in the preference relation, $\preceq^i$, assuming both players are rational.

This model does not use variable payoff functions, so $\mathcal{V} \triangleq \{null\}$. There are also no constraints to be applied in the game model.

7.4.2 Identification of States and Transitions.

For Step 2: Identification of States and Transitions of the temporal hypergame framework, it is necessary to define the rules of the game the players must follow and is derived from the game tree. The game arena is given by:

$$\Phi_{net}^T \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \Upsilon, \twoheadrightarrow, \gamma, \mathcal{V}, \chi)$$

W consists of sixteen game states, including start, D_{ND}, D_{DE}, D_{PR}, D_S, $D_{ND}A_{NA}$, $D_{ND}A_{AT}$, $D_{ND}A_{ZD}$, $D_{DE}A_{NA}$, $D_{DE}A_{AT}$, $D_{DE}A_{ZD}$, $D_{PR}A_{NA}$, $D_{PR}A_{AT}$, $D_{PR}A_{ZD}$, $D_S A_{NA}$, $D_S A_{AT}$, $D_S A_{ZD}$. The function $\longrightarrow$ defines the game transitions such that (W x Σ) $\to$ W. The possible transitions are:

- start x ND $\to D_{ND}$ where ND $\in \Sigma_D$

- start x DE $\rightarrow D_{DE}$ where DE $\in \Sigma_D$

- start x PR $\rightarrow D_{PR}$ where PR $\in \Sigma_D$

- start x S $\rightarrow D_S$ where S $\in \Sigma_D$

- D_{ND} x NA $\rightarrow D_{ND}A_{NA}$ where NA $\in \Sigma_A$

- D_{ND} x AT $\rightarrow D_{ND}A_{AT}$ where AT $\in \Sigma_A$

- D_{ND} x ZD $\rightarrow D_{ND}A_{ZD}$ where ZD $\in \Sigma_A$

- D_{DE} x NA $\rightarrow D_{DE}A_{NA}$ where NA $\in \Sigma_A$

- D_{DE} x AT $\rightarrow D_{DE}A_{AT}$ where AT $\in \Sigma_A$

- D_{DE} x ZD $\rightarrow D_{DE}A_{ZD}$ where ZD $\in \Sigma_A$

- D_{PR} x NA $\rightarrow D_{PR}A_{NA}$ where NA $\in \Sigma_A$

- D_{PR} x AT $\rightarrow D_{PR}A_{AT}$ where AT $\in \Sigma_A$

- D_{PR} x ZD $\rightarrow D_{PR}A_{ZD}$ where ZD $\in \Sigma_A$

- D_S x NA $\rightarrow D_S A_{NA}$ where NA $\in \Sigma_A$

- D_S x AT $\rightarrow D_S A_{AT}$ where AT $\in \Sigma_A$

- D_S x ZD $\rightarrow D_S A_{ZD}$ where ZD $\in \Sigma_A$

- $D_{ND}A_{NA} \rightarrow$ start

- $D_{ND}A_{AT} \rightarrow$ start

- $D_{ND}A_{ZD} \rightarrow$ start

- $D_{DE}A_{NA} \rightarrow$ start

- $D_{DE}A_{AT} \rightarrow$ start

- $D_{DE}A_{ZD} \rightarrow$ start

- $D_{PR}A_{NA} \rightarrow$ start

- $D_{PR}A_{AT} \rightarrow$ start

- $D_{PR}A_{ZD} \rightarrow$ start

- $D_{S}A_{NA} \rightarrow$ start

- $D_{S}A_{AT} \rightarrow$ start

- $D_{S}A_{ZD} \rightarrow$ start

The initial state of the game w_0, is equal to start $\in$ W, where $\overrightarrow{start} = \{D_{ND}, D_{DE}, D_{PR}, D_S\}$. The belief context β_0 is set to 0.8 (which is selected randomly for this example), the update function for the belief context $\hookrightarrow$ simply makes no modification to the belief contexts (i.e. $\beta 0 \hookrightarrow \beta_{new}$ implies $\beta 0 = \beta_{new}$). The CMS $\mathcal{C}$, and the initial value $\mathcal{C}_0$ is set to the NEMS value for the hypergame. The fear-of-being-outguessed Υ, is set to an initial value γ of zero. The update function $\twoheadrightarrow$ maps $\gamma \twoheadrightarrow \Upsilon = 0$. Therefore there is no update. The χ function assigns he player whose turn it is to the game state w $\in$ W where W $\rightarrow$ N. For example, $\chi(start) = $ D, while $\chi(D_{ND}) = \chi(D_{DE}) = \chi(D_{PR}) = \chi(D_S) = $ A.

7.4.3 Game Mapping.

For Step 3: Game Mapping, an extensive form game tree $\mathcal{T}$ is associated with the network game arena, ϕ_{net}^T. The extensive form game tree is defined as:

$$\mathcal{T} = (S, \Rightarrow, s_0, \lambda) \tag{7.1}$$

where (S, $\Rightarrow$) is a countably infinite tree rooted at s_0 with edges from Σ. The nodes of the tree are given by S, where S = {root, D_{ND}, D_{DE}, D_{PR}, D_S, $D_{ND}A_{NA}$, $D_{ND}A_{AT}$, $D_{ND}A_{ZD}$, $D_{DE}A_{NA}$, $D_{DE}A_{AT}$, $D_{DE}A_{ZD}$, $D_{PR}A_{NA}$, $D_{PR}A_{AT}$, $D_{PR}A_{ZD}$, D_SA_{NA}, D_SA_{AT}, D_SA_{ZD}}. The root of the tree denoted s_0, is equal to root $\in$ S. The $\overset{x}{\Rightarrow}$ is the function that moves between nodes of the tree using the edge denoted by x $\in \Sigma$.

The possible nodes transitions are {root $\overset{ND}{\Longrightarrow} D_{ND}$, root $\overset{DE}{\Longrightarrow} D_{DE}$, root $\overset{PR}{\Longrightarrow} D_{PR}$, root $\overset{S}{\Rightarrow} D_S$ where ND, DE, PR, S $\in \sigma_D$} $\cup$ {$D_{ND} \overset{NA}{\Longrightarrow} D_{ND}A_{NA}$, $D_{ND} \overset{AT}{\Longrightarrow} D_{ND}A_{AT}$, $D_{ND} \overset{ZD}{\Longrightarrow} D_{ND}A_{ZD}$, $D_{DE} \overset{NA}{\Longrightarrow} D_{DE}A_{NA}$, $D_{DE} \overset{AT}{\Longrightarrow} D_{DE}A_{AT}$, $D_{DE} \overset{ZD}{\Longrightarrow} D_{DE}A_{ZD}$, $D_{PR} \overset{NA}{\Longrightarrow} D_{PR}A_{NA}$, $D_{PR} \overset{AT}{\Longrightarrow} D_{PR}A_{AT}$, $D_{PR} \overset{ZD}{\Longrightarrow} D_{PR}A_{ZD}$ $D_S \overset{NA}{\Longrightarrow} D_SA_{NA}$, $D_S \overset{AT}{\Longrightarrow} D_SA_{AT}$, $D_S \overset{ZD}{\Longrightarrow} D_SA_{ZD}$ where NA, AT, ZD $\in \sigma_A$}.

The function λ is S $\rightarrow$ W, where

- $\lambda(s_0) = w_0$

- $\forall s, s' \in$ S, if s $\overset{a}{\Rightarrow} s'$ then $\lambda(s) \overset{a}{\Rightarrow} \lambda(s')$

- if $\lambda(s) =$ w and w $\overset{a}{\rightarrow} w'$ there exists $s' \in$ S s.t. s $\overset{a}{\Rightarrow} s'$ and $\lambda(s') = w'$

7.4.4 Path Structuring.

For Step 4: Path Structuring, the players individual paths through the game tree are defined. Theses paths are later used to define the strategies, but are not required. Defining the paths reduces the amount of notation required for the strategy definitions. The model for the network temporal hypergame can be represented by $M_{net} = (\mathcal{T}, \mathcal{V})$. The game tree, $\mathcal{T}$ is given from the previous game mapping and the valuation function $\sqsubseteq$ is given by:

- $\mathcal{V}(p_{int}) = \{s_0\}$

- $\mathcal{V}(p_{domA}) = \{D_{ND}, D_{ND}A_{AT}\}$ or $\{D_{ND}, D_{ND}A_{ZD}\}$ or $\{D_{DE}, D_{DE}A_{AT}\}$ or $\{D_{DE}, D_{DE}A_{ZD}\}$

167

- $\mathcal{V}(p_{domD_{exp}}) = \{D_{DE}\}$

- $\mathcal{V}(p_{worst_A}) = \{D_{ND}, D_{ND}A_{NA}\}$ or $\{D_{DE}, D_{DE}A_{NA}\}$

- $\mathcal{V}(p_{worst_D}) = \{D_{ND}, D_{ND}A_{AT}\}$ or $\{D_{ND}, D_{ND}A_{ZD}\}$

When the hypergame advantageS (defender can Provide Ruse and Shutdown) is considered, the additional functions are included; while the other functions remain the same:

- $\mathcal{V}(p_{domD_{hyp}}) = \{D_{PR}, D_{PR}A_{AT}\}$

- $\mathcal{V}(p_{worstA_{hyp}}) = \{D_S, D_S A_{AT}\}$ or $\{D_S, D_S A_{ZD}\}$

The $p_{domD_{hyp}}$ represents the dominant path for the defender from the attacker's point-of-view. The attacker has failed to account for the defender's provide ruse option. While the $p_{domD_{exp}}$ represents the dominant path for the defender when the defender's advantage (Provide Ruse) is considered. The attacker is unaware of this option, and therefore assumes $p_{domD_{exp}}$ is the expected outcome for a rational defender.

7.4.5 Define Player Strategies.

For Step 5: Define Player Strategies, strategies for each player are defined in terms of the states and transitions, as well as the paths defined previously. In this game the attacker's strategy can be defined as:

$$Strat^A \equiv ([p_{int} \mapsto ND]^D \Rightarrow [p_{domD_{exp}} \mapsto AT]^A \vee [p_{domD_{exp}} \mapsto ZD]^A) \cdot$$
$$([p_{int} \mapsto DE]^D \Rightarrow [p_{domD_{exp}} \mapsto AT]^A \vee [p_{domD_{exp}} \mapsto ZD]^A) \cdot ([p_{int} \mapsto ND]^D \Rightarrow$$
$$[p_{worstA_{exp}} \mapsto NA]^A) \cdot ([p_{int} \mapsto DE]^D \Rightarrow [p_{worstA_{exp}} \mapsto NA]^A)$$

From the attacker's perspective, the defender's strategy is defined as:

$$Strat^D_A \equiv ([p_{int} \mapsto ND]^D) \cdot ([p_{int} \mapsto DE]^D) \cdot ([p_{int} \mapsto ND]^D \Rightarrow p_{domD_{exp}}) \cdot$$
$$([p_{int} \mapsto DE]^D \Rightarrow p_{domD_{exp}})$$

But the defender has two extra actions available (Provide Ruse and Shutdown) that the attacker is not aware of, so the defender's strategy can be defined as:

$$Strat^D \equiv ([p_{int} \mapsto ND]^D) \cdot ([p_{int} \mapsto DE]^D) \cdot ([p_{int} \mapsto PR]^D \cdot ([p_{int} \mapsto S]^D) \cdot$$

$$([p_{int} \mapsto ND]^D \Rightarrow p_{domD_{exp}}) \cdot ([p_{int} \mapsto DE]^D \Rightarrow p_{domD_{exp}}) \cdot ([p_{int} \mapsto PR]^D \Rightarrow$$

$$[p_{domD_{hyp}} \mapsto AT]^A) \cdot ([p_{int} \mapsto S]^D \Rightarrow [p_{worstA_{hyp}} \mapsto AT]^A \vee [p_{worstA_{hyp}} \mapsto ZD]^A)$$

7.4.6   Analyze Model.

For the final Step 6: Analyze Model, the previous definitions are used to logical analysis the constructed model. From the previous definitions, $Strat_A^D \rightsquigarrow_D (p_{domD_{exp}} \vee p_{worst_D})$, which means the defender can either follow the dominant strategy or the worst case strategy from the view of the attacker. A rational defender is always assumed to follow the dominant strategy. In this case if the defender plays "Defend" then the defender can ensure the worst outcome is avoided:

$$[turn_D \mapsto D]^D \rightsquigarrow_D \neg p_{worst_D}$$

This represents the result of the game where the defender always desires to defend. With this, the defender would expect the attacker to reach the expected best case p_{expA}. Which means the defender would expect $Strat^A \rightsquigarrow_A p_{domA}$.

The attacker follows $[turn_A \mapsto AT]^A \wedge [turn_A \mapsto ZD]^A$. This allows the attacker to achieve the best outcome, while allowing the defender to follow their perceived dominant strategy.

For the defender the solution is always to use the "Defend" strategy:

$$[turn_D \mapsto D]^D \rightsquigarrow_D \neg p_{worst_D}$$

The attacker's strategy is to always choose the Attack (AT) or Zero-Day (ZD).

$$[turn_A \mapsto AT]^A \wedge [turn_A \mapsto ZD]^A \rightsquigarrow_A p_{domA}$$

Up to this point the analysis has focused on the solution to the game without considering the hypergame advantage for the defender. From the previous definitions, $Strat^D \leadsto_D (p_{domD_{hyp}} \lor p_{worst_D})$, which means the defender can either follow the hypergame strategy or the worst case strategy. It is assumed that a rational defender always plays to the hypergame strategy, since it is dominant. In this case, if the defender plays "Defend" or the hidden hypergame strategies Provide Ruse and Shutdown, then the defender can ensure the worst outcome is avoided:

$$[turn_D \mapsto ND]^D \leadsto_D \neg p_{worst_D}$$

This shows the hypergame result of the game where the defender would expect the defender to reach the expected worst case p_{worst_D}.

The attacker still follows $[turn_A \mapsto AT]^A \land [turn_A \mapsto ZD]^A$. This allows the attacker to achieve their perceived best outcome, which is $Strat^A \leadsto_A p_{domA}$ from the previous analysis. Because of the hypergame, the attacker does not end up in the dominant or worst outcome $Strat^A \neg \leadsto_A p_{worstA_{hyp}} \lor p_{domA}$.

Since the solution to the hypergame is for the defender to play Provide Ruse and the attacker to play Attack or Zero-Day, the defender can guarantee the worst possible outcome is avoided.

$$[turn_D \mapsto PR]^D \leadsto_D \neg p_{worst_D} \land (p_{domD_{hyp}}$$

The attacker's strategy is to always play the action or Attack or Zero-Day.

$$[turn_A \mapsto AT]^A \land [turn_A \mapsto ZD]^A$$

The temporal hypergame analysis shows that the defender can improve upon their outcome by leveraging the attacker's misperception. It also shows that the defender can guarantee the attacker does not reach the dominant (best case) outcome.

7.5 Cyber Physical Security Application

Cyber Physical Systems (CPS) are physical entities controlled and monitored by computer-based algorithms. A CPS is usually designed as a network of interacting systems with inputs and outputs as opposed to standalone devices. CPS is similar in architecture to the Internet of Things (IoT), but require higher coordination between physical and computational elements. An overview of CPS is provided in Figure 7.6. The specific application focus of this research (shown in Figure 7.7) is on the cyber security of CPSs.
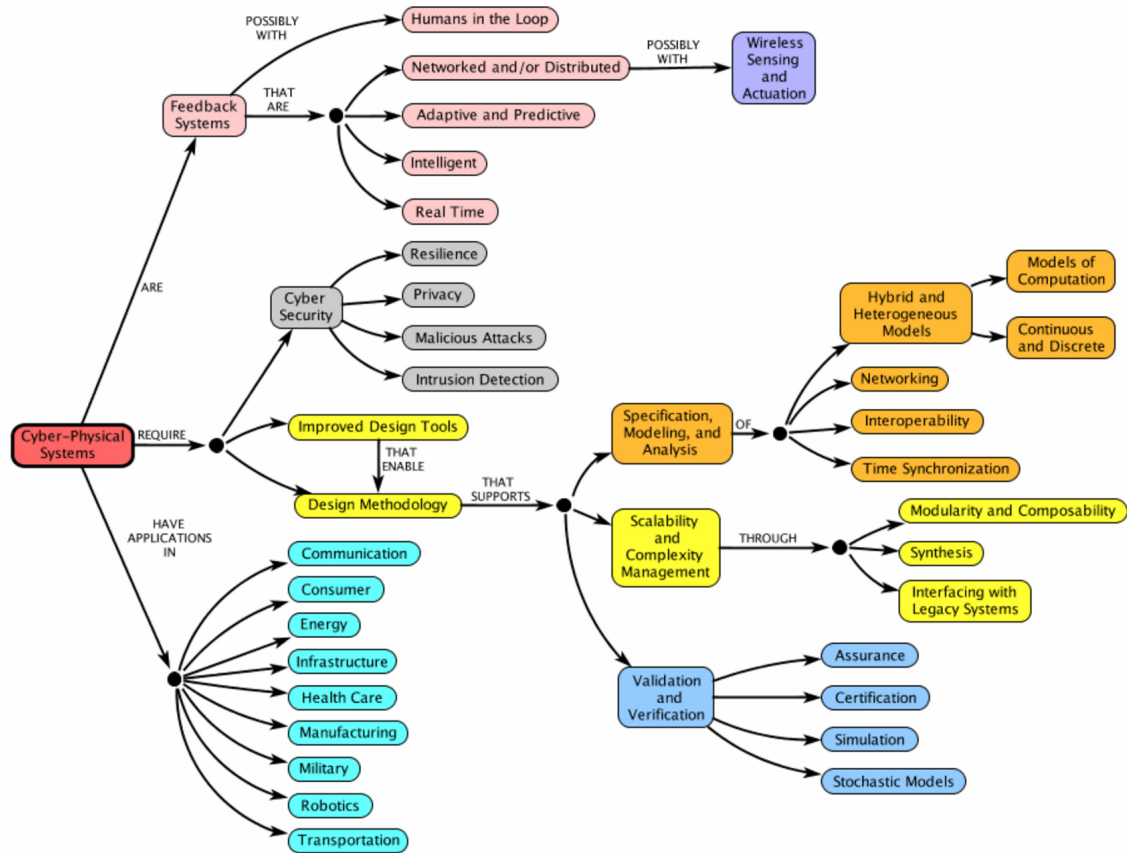
Figure 7.6: Overview of Cyber Physical Systems [66].
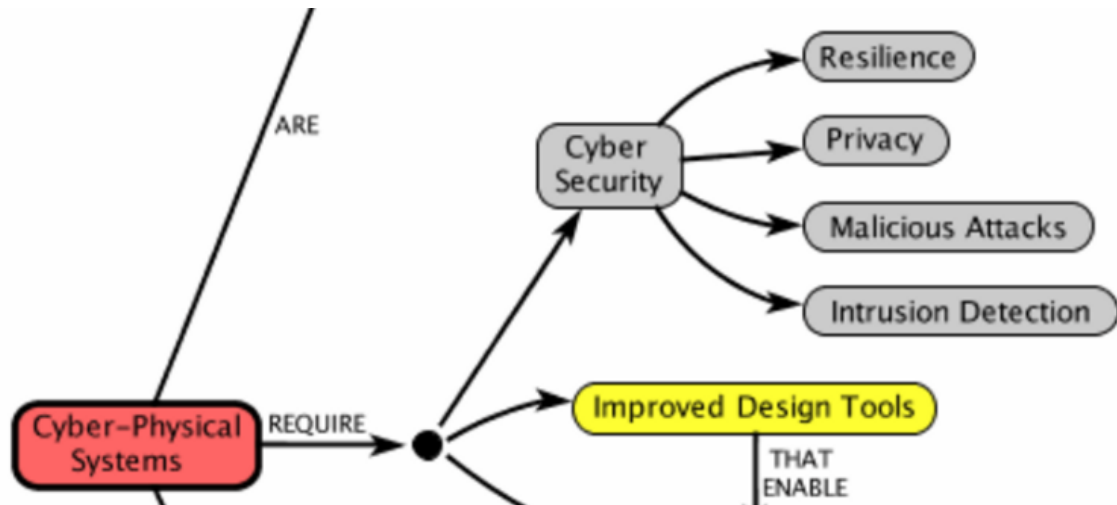
Figure 7.7: Focus Area of Cyber Physical System Application [66].

These types of systems include those found in avionics, ships, satellites, cars and Supervisory Control and Data Acquisition) SCADA systems. Reports on exploits such as Stuxnet [323] and APT28 [99] are appearing more frequently and are becoming a growing concern (see Section 1.1.2 in Chapter 1 for a more detailed listing of exploits). As CPSs begin to depend more on meta-level information infrastructures and as this IP- based technology begins to be integrated into these systems, critical CPSs are becoming a larger part of the already established vulnerability target. This puts the mission assurance of these systems at risk with potentially higher stakes than losing data or intellectual property. A Supervisory Control and Data Acquisition (SCADA) system is an example of a Cyber Physical Systems (CPS) and is a type of Industrial Control System (ICS). SCADA systems monitor and control industrial processes that exist in the physical world.

7.5.1 Cyber Physical SCADA Security Hypergame Example - Game Theoretic.

Hewett et al. [165, 307] present a game theoretic model for cyber-security analysis of SCADA systems. The game is based on the classic attacker-defender game. The authors present the game in extensive form as shown in Figure 7.8. The defender has two actions: defend the SCADA network or not defend. The attacker has five actions: sybil (identity spoofing), node compromise (control of a sensor node), eavesdropping (traffic sniffing), data injection (datastore or communication channels), or no attack.
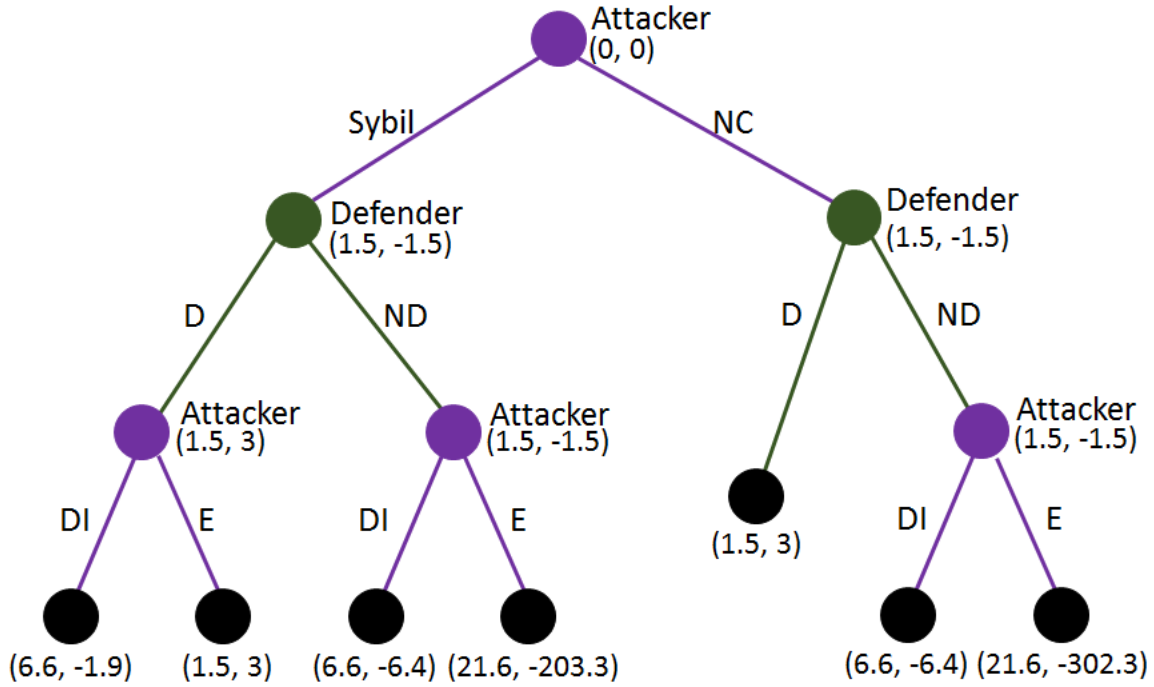
Figure 7.8: SCADA Sensor Network Game [165, 307].

The unique property of the game is that it is not symmetric. Depending on whether the attacker chooses sybil or node compromise as a first action, leads to the set of actions that are available to the attacker in the second round. If the attacker chooses sybil then in the next round the attacker chooses from the actions of

eavesdropping or data injection. For node compromise, the attacker chooses from the actions of eavesdropping or data injection. The authors use variables in the payoff functions. The payoff function is denoted $U_d(p, a)$, of player p at a decision node of depth d as a result of action a. Action a can be either taken by player p or the opponent. The payoff function (utility) is calculated by :

$$U_d(p, a) = U_{d-1}(p, a') + B(p, a, d) \tag{7.2}$$

$B(p, a, d)$ denotes the behavior of the impact of the action and a' denotes the action of p's opponent. At the root of the game tree (i.e. start of the game) there are no previous payoffs, so the initial payoff is $U_0(p, nil) = (0, 0)$ where nil is no action. $B(p, a, d)$ is the behavior of the impact of the action, as shown in Table 7.1. It depends on the action a, the player p, and the depth d of the game tree as an indicator of the game's advancement.

Table 7.1: SCADA Behavior of the Impact [165, 307].

$B(p, a, d)$	a is A's action	a is D's action
p = A (Attacker)	$d * Impact(a)$	0
p = D (Defender)	$-Impact(a)^d$	$Impact(a)$

The impact function shown in Table 7.1, determines the impact of confidentiality, integrity, and availability. Let C(a), I(a), and A(a) be the confidentiality, integrity, and availability with the corresponding weights w_C, w_I, and w_A of action a. The function is defined as:

$$Impact(a) = w_C C(a) + w_I I(a) + w_A A(a) \tag{7.3}$$

The weight sum to 1 and are fixed at $w_C = 0.1$, $w_I = 0.6$, and $w_A = 0.3$. The impact function is calculated according to Table 7.2. Not defend and not attack have no impact on the model and are excluded from the table. In Table 7.2, the value 1 is considered low, 4 is moderate, and 8 is high, in terms of impact.

Table 7.2: SCADA Model Impact Function [165, 307].

Description	C(a)	I(a)	A(a)	Impact(a)
Defend	6	6	1	4.5
Sybil	6	1	1	1.5
Node Compromise	6	1	1	1.5
Eavesdropping	8	1	1	1.7
Data Injection	1	8	6	6.7

7.5.1.1 Initial Game Definition.

To apply Step 1: Initial Game Definition of the temporal hypergame framework to the information from the SCADA model, it is necessary to create a temporal hypergame model. Let the temporal hypergame be:

$$H_{SCADA}^T \triangleq \{G_D^T\}$$

The perceived game G_{SCADA}^T is:

$$G_{SCADA}^T \triangleq (N, \Sigma_{Full}, \Phi^T, \{\preceq^i\}_{i \in N})$$

Where N = {Attacker (A), Defender (D)} is the set of players. Σ_{Full} is the finite set of player actions such that $\Sigma_{Full} = \Sigma_A \cup \Sigma_D$, $\Sigma_A = \{Sybil(Sy), NodeCompromise(NC),$ $Eavesdropping(E), DataInjection(DI), NotAttack(nil)\}$, and $\Sigma_D = \{Defend(D),$ $NotDefend(nil)\}$.

175

The preference ordering function for the i_{th} player $\preceq^i = u_i(x,y) \leq u_i(x',y)$ for $i \in N$. This means for the attacker $\preceq^A$ indicates the outcome (x,y) is preferred to outcome (x',y) if $x \leq x' \; \forall (x,y) \in W$. For the defender $\preceq^D$ indicates the outcome (x,y) is preferred to outcome (x,y') if $y \leq y' \; \forall (x,y) \in W$. The Nash equilibrium concept is encoded in the preference relation, $\preceq^i$, assuming both players are rational.

The variable payoff function is denoted by:

$$\mathcal{V} \triangleq \{\omega, \omega_{init}, \psi, \delta\}$$

The payoff function variables are given by $\omega = \{w_C, w_I, w_A, d, Impact(a)\}$. The initial values ω_{init}, are set to 0 except for $w_C = 0.1$, $w_I = 0.6$, and $w_A = 0.3$. The function ψ works as follows:

- If variable Impact(a)

    - If a = Defend, then Impact = 4.5

    - if a = Sybil, then Impact = 1.5

    - if a = Node Compromise, then Impact = 1.5

    - if a = Eavesdropping, then Impact = 1.7

    - if a = Data Injection, then Impact = 6.7

- if variable d, then d = d + 1

With the proposed model there are no constraints to be applied.

7.5.1.2   Identification of States and Transitions.

For Step 2: Identification of States and Transitions of the temporal hypergame framework, it is necessary to define the rules of the game the players must follow and is derived from the game tree. The game arena is given by:

$$\Phi_{SCADA}^T \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \Upsilon, \twoheadrightarrow, \gamma, \mathcal{V}, \chi)$$

176

W consists of sixteen game states, including start, A_{Sy}, A_{NC}, $A_{Sy}D_D$, $A_{Sy}D_{nil}$, $A_{NC}D_D$, $A_{NC}D_{nil}$, $A_{ISE}D_D$, $A_{Sy}D_DA_E$, $A_{Sy}D_DA_{nil}$, $A_{Sy}D_{nil}A_E$, $A_{Sy}D_{nil}A_{DI}$, $A_{Sy}D_DA_E$, $A_{Sy}D_DA_{DI}$, $A_{Sy}D_DA_{nil}$, $A_{NC}D_{nil}A_E$, and $A_{NC}D_{nil}A_{DI}$. The function $\longrightarrow$ defines the game transitions such that (W x Σ) $\rightarrow$ W. The possible transitions are:

- start x Sy $\rightarrow A_{Sy}$ where Sy $\in \Sigma_A$

- start x NC $\rightarrow A_{NC}$ where NC $\in \Sigma_A$

- A_{Sy} x D $\rightarrow A_{Sy}D_D$ where D $\in \Sigma_D$

- A_{Sy} x nil $\rightarrow A_{Sy}D_{nil}$ where nil $\in \Sigma_D$

- A_{NC} x D $\rightarrow A_{NC}D_D$ where D $\in \Sigma_D$

- A_{NC} x nil $\rightarrow A_{NC}D_{nil}$ where nil $\in \Sigma_D$

- $A_{Sy}D_D$ x E $\rightarrow A_{Sy}D_DA_E$ where E $\in \Sigma_A$

- $A_{Sy}D_D$ x nil $\rightarrow A_{Sy}D_DA_{nil}$ where nil $\in \Sigma_A$

- $A_{Sy}D_{nil}$ x E $\rightarrow A_{Sy}D_{nil}A_E$ where E $\in \Sigma_A$

- $A_{Sy}D_{nil}$ x DI $\rightarrow A_{Sy}D_{nil}A_{DI}$ where DI $\in \Sigma_A$

- $A_{NC}D_D$ x E $\rightarrow A_{Sy}D_DA_E$ where E $\in \Sigma_A$

- $A_{NC}D_D$ x DI $\rightarrow A_{Sy}D_DA_{DI}$ where DI $\in \Sigma_A$

- $A_{NC}D_D$ x nil $\rightarrow A_{Sy}D_DA_{nil}$ where nil $\in \Sigma_A$

- $A_{NC}D_{nil}$ x E $\rightarrow A_{NC}D_{nil}A_E$ where E $\in \Sigma_A$

- $A_{NC}D_{nil}$ x DI $\rightarrow A_{NC}D_{nil}A_{DI}$ where DI $\in \Sigma_A$

The initial state of the game w_0, is equal to start $\in$ W, where $\overrightarrow{start}$ = $\{A_{Sy}, A_{NC}, A_{ISE}\}$. The belief context β_0 is set to 0.8 (which is randomly chosen for this example), the update function for the belief context $\hookrightarrow$ simply makes no modification to the belief contexts (i.e. $\beta 0 \hookrightarrow \beta_{new}$ implies $\beta 0 = \beta_{new}$). The CMS $\mathcal{C}$, and the initial value $\mathcal{C}_0$ is set to the NEMS value for the hypergame. The fear-of-being-outguessed Υ, is set to an initial value γ of zero. The update function $\twoheadrightarrow$ maps $\gamma \twoheadrightarrow \Upsilon = 0$. Therefore there is no update. The χ function assigns the player whose turn it is to the game state w $\in$ W where W $\rightarrow$ N. For example, $\chi(start) = $ A, while $\chi(A_C) = \chi(A_D) = $ D.

7.5.1.3 Game Mapping.

For Step 3: Game Mapping, an extensive form game tree $\mathcal{T}$ is associated with the SCADA security game arena, ϕ_{SCADA}^T. The extensive form game tree is defined as:

$$\mathcal{T} = (S, \Rightarrow, s_0, \lambda)$$

where (S, $\Rightarrow$) is a countably infinite tree rooted at s_0 with edges from Σ. The nodes of the tree are given by S, where S = {root, A_{Sy}, A_{NC}, $A_{Sy}D_D$, $A_{Sy}D_{nil}$, $A_{NC}D_D$, $A_{NC}D_{nil}$, $A_{Sy}D_D A_E$, $A_{Sy}D_D A_{nil}$, $A_{Sy}D_{nil}A_E$, $A_{Sy}D_{nil}A_{DI}$, $A_{Sy}D_D A_E$, $A_{Sy}D_D A_{DI}$, $A_{Sy}D_D A_{nil}$, $A_{NC}D_{nil}A_E$, and $A_{NC}D_{nil}A_{DI}$}. The root of the tree denoted s_0, is equal to root $\in$ S. The $\overset{x}{\Rightarrow}$ is the function that moves between nodes of the tree using the edge denoted by x $\in \Sigma$.

The possible nodes transitions are {root $\overset{Sybil}{\Longrightarrow} A_{Sy}$, root $\overset{NodeCompromise}{\Longrightarrow} A_{NC}$ where Sybil, Node Compromise $\in \sigma_A$} $\cup \{A_{Sy} \overset{D}{\Rightarrow} A_{Sy}D_D, A_{Sy} \overset{nil}{\Rightarrow} A_{Sy}D_{nil}, A_{NC} \overset{D}{\Rightarrow} A_{NC}D_D, A_{NC} \overset{nil}{\Rightarrow} A_{NC}D_{nil}$ where D, nil $\in \sigma_D$} $\cup \{A_{Sy}D_D \overset{E}{\Rightarrow} A_{Sy}D_D A_E, A_{Sy}D_D \overset{nil}{\Rightarrow} A_{Sy}D_D A_{nil}, A_{Sy}D_{nil} \overset{E}{\Rightarrow} A_{Sy}D_{nil}A_E, A_{Sy}D_{nil} \overset{DI}{\Rightarrow} A_{Sy}D_{nil}A_{DI}, A_{NC}D_D \overset{E}{\Rightarrow} A_{NC}D_D A_E, A_{NC}D_D \overset{DI}{\Rightarrow} A_{NC}D_D A_{DI}, A_{NC}D_D \overset{nil}{\Rightarrow} A_{NC}D_D A_{nil}, A_{NC}D_{nil} \overset{E}{\Rightarrow} A_{NC}D_{nil}A_E, A_{NC}D_{nil} \overset{DI}{\Rightarrow} A_{NC}D_{nil}A_{DI}$ where E, DI, nil $\in \sigma_A$}.

The function λ is S $\rightarrow$ W, where

- $\lambda(s_0) = w_0$

- $\forall s, s' \in S$, if s $\overset{a}{\Rightarrow}$ s' then $\lambda(s) \overset{a}{\Rightarrow} \lambda(s')$

- if $\lambda(s)$ = w and w $\overset{a}{\rightarrow}$ w' there exists $s' \in$ S s.t. s $\overset{a}{\Rightarrow}$ s' and $\lambda(s')$ = w'

7.5.1.4   Path Structuring.

For Step 4: Path Structuring, the players individual paths through the game tree are defined. Theses paths are later used to define the strategies, but are not required. Defining the paths reduces the amount of notation required for the strategy definitions. The model for the SCADA security temporal hypergame can be represented by $M_{SCADA} = (\mathcal{T}, \mathcal{V})$. The game tree, $\mathcal{T}$ is given from the previous game mapping and the valuation function $\mathcal{V}$ is given by:

- $\mathcal{V}(p_{int}) = \{s_0\}$

- $\mathcal{V}(p_{dom}) = \{A_{Sy}, A_{Sy}D_D A_E\}$

- $\mathcal{V}(p_{worst_A}) = \{A_{NC}, A_{NC}D_D\}$

- $\mathcal{V}(p_{worst_D}) = \{A_{NC}, A_{NC}D_{nil}\}$ or $\{A_{Sy}, A_{Sy}D_{nil}\}$

7.5.1.5   Define Player Strategies.

For Step 5: Define Player Strategies, strategies for each player are defined in terms of the states and transitions, as well as the paths defined previously. In this game the Attacker's strategy can be defined as:

$$Strat^A \equiv ([p_{int} \mapsto Sy]^A) \cdot ([p_{int} \mapsto NC]^A) \cdot ([p_{dom} \mapsto D]^D \Rightarrow [p_{max_A} \mapsto E]^A) \cdot$$
$$([p_{dom} \mapsto nil]^D \Rightarrow [p_{max_A} \mapsto DI]^A)$$

The Defender's strategy can be defined as:

$$Strat^D \equiv ([p_{int} \mapsto Sy]^A \Rightarrow [p_{dom} \mapsto D]^D) \cdot ([p_{int} \mapsto NC]^A \Rightarrow [p_{dom} \mapsto D]^D) \cdot$$

$$([p_{int} \mapsto Sy]^A \Rightarrow [p_{worst_D} \mapsto nil]^D) \cdot ([p_{int} \mapsto NC]^A \Rightarrow [p_{worst_D} \mapsto nil]^D)$$

7.5.1.6 Analyze Model.

For the final Step 6: Analyze Model, the previous definitions are used to logical analysis the constructed model. From the previous definitions, $Strat^D \rightsquigarrow_D (p_{domD} \vee p_{worst_D})$, which means the defender can either follow the dominate strategy or the worst case strategy. If the defender plays "Defend" then the defender can ensure the worst outcome is avoided:

$$[turn_D \mapsto D]^D \rightsquigarrow_D \neg p_{worst_D}$$

This indicates the standard result of the game where the defender always desires to defend. With this, the defender would expect the attacker to reach the expected best case p_{expA}. Which means the defender expects the attacker to follow $Strat^A \rightsquigarrow_A p_{expA}$.

For the defender the strategy is to always defend:

$$[turn_D \mapsto D]^D \rightsquigarrow_D \neg p_{worst_D}$$

7.5.2 Cyber Physical SCADA Security Temporal Hypergame Example - Attacker View.

This section extends the previous game theoretic SCADA security example with additional actions that can lead to misperceptions, in order to validate more of the temporal hypergame framework. This example is designed to show how misperceptions can lead to different results based on the view of the attacker. The payoff function is denoted $U_d(p, a)$, of player p at a decision node of depth d as a result of action a. Action a can be either taken by player p or the opponent. The payoff function (utility) is calculated by :

$$U_d(p, a) = U_{d-1}(p, a') + B(p, a, d) \tag{7.4}$$

$B(p, a, d)$ denotes the behavior of the impact of the action and a' denotes the action of p's opponent. At the root of the game tree (i.e. start of the game) there are no previous payoffs, so the initial payoff is $U_0(p, nil) = (0, 0)$ where nil is no action. $B(p, a, d)$ is the behavior of the impact of the action, as shown in Table 7.3. It depends on the action a, the player p, and the depth d of the game tree as an indicator of the game's advancement.

Table 7.3: SCADA Behavior of the Impact [165, 307].

$B(p, a, d)$	a is A's action	a is D's action
p = A (Attacker)	$d * Impact(a)$	0
p = D (Defender)	$-Impact(a)^d$	$Impact(a)$

The impact function shown in Table 7.1, determines the impact of confidentiality, integrity, and availability. Let C(a), I(a), and A(a) be the confidentiality, integrity, and availability with the corresponding weights w_C, w_I, and w_A of action a. The function is defined as:

$$Impact(a) = w_C C(a) + w_I I(a) + w_A A(a) \tag{7.5}$$

In order to make the game more interesting for hypergame modeling, the SCADA impact function is expanded by adding two new actions - infect support equipment and ruin/hide. The updated impact function is shown in Table 7.4. The infect support equipment action is only available to the attacker in the first round (the attacker's initial decision). The ruin/hide action is only available to the attacker if the attacker chose infect support equipment in the first round. The weight sum to 1 and are fixed

at $w_C = 0.1$, $w_I = 0.6$, and $w_A = 0.3$. Not defend and not attack have no impact on the model and are excluded from the table. In Table 7.4, the value 1 is considered low, 4 is moderate, and 8 is high, in terms of impact.

Table 7.4: Updated SCADA Model Impact Function.

Description	C(a)	I(a)	A(a)	Impact(a)
Defend	6	6	1	4.5
Sybil	6	1	1	1.5
Node Compromise	6	1	1	1.5
Eavesdropping	8	1	1	1.7
Data Injection	1	8	6	6.7
Infect Support Equipment	6	3	3	3.3
Ruin/Hide	2	5	3	4.1

The simple expanded SCADA sensor network game is shown in Figure 7.9. This shows the addition of the extra Infect Support Equipment (ISE) option for the attacker in the first round of play.

7.5.2.1 Initial Game Definition.

To apply Step 1: Initial Game Definition of the temporal hypergame framework to the information from the SCADA model, it is necessary to create a temporal hypergame model. Let the temporal hypergame be:

$$H_{SCADA}^T \triangleq \{G_D^T\}$$

The perceived game G_{SCADA}^T is:

$$G_{SCADA}^T \triangleq (N, \Sigma_{Full}, \Phi^T, \{\preceq^i\}_{i \in N})$$
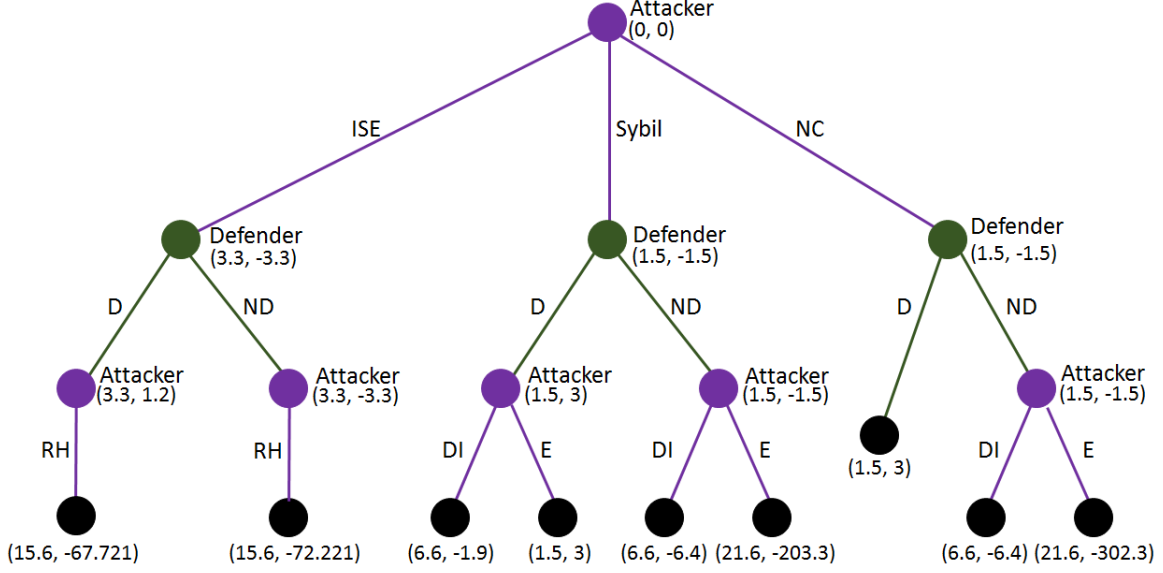
Figure 7.9: Expanded SCADA Sensor Network Game.

Where N = {Attacker (A), Defender (D)} is the set of players. Σ_{Full} is the finite set of player actions such that $\Sigma_{Full} = \Sigma_A \cup \Sigma_D$, $\Sigma_A = \{Sybil(Sy), NodeCompromise(NC),$ $Eavesdropping(E), DataInjection(DI), NotAttack(nil), InfectSupportEquipment(ISE),$ $Ruin/Hide(RH)\}$, and $\Sigma_D = \{Defend(D), NotDefend(nil)\}$.

The preference ordering function for the i_{th} player $\preceq^i = u_i(x, y) \leq u_i(x', y)$ for $i \in$ N. This means for the attacker $\preceq^A$ indicates the outcome (x,y) is preferred to outcome (x', y) if $x \leq x'$ $\forall (x, y) \in W$. For the defender $\preceq^D$ indicates the outcome (x,y) is preferred to outcome (x, y') if $y \leq y'$ $\forall (x, y) \in W$. The Nash equilibrium concept is encoded in the preference relation, $\preceq^i$, assuming both players are rational.

The variable payoff function is denoted by:

$$\mathcal{V} \triangleq \{\omega, \omega_{init}, \psi, \delta\}$$

The payoff function variables are given by $\omega = \{w_C, w_I, w_A, d, Impact(a)\}$. The initial values ω_{init}, are set to 0 except for $w_C = 0.1$, $w_I = 0.6$, and $w_A = 0.3$. The function ψ works as follows:

183

- If variable Impact(a)

  - If a = Defend, then Impact = 4.5

  - if a = Sybil, then Impact = 1.5

  - if a = Node Compromise, then Impact = 1.5

  - if a = Eavesdropping, then Impact = 1.7

  - if a = Data Injection, then Impact = 6.7

  - if a = Infect Support Equipment, then Impact = 3.3

  - if a = Ruin/Hide, then Impact = 4.1

- if variable d, then d = d + 1

With the proposed model there are no constraints to be applied.

7.5.2.2 Identification of States and Transitions.

For Step 2: Identification of States and Transitions of the temporal hypergame framework, it is necessary to define the rules of the game the players must follow and is derived from the game tree. The game arena is given by:

$$\Phi^T_{SCADA} \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \Upsilon, \twoheadrightarrow, \gamma, \mathcal{V}, \chi)$$

W consists of sixteen game states, including start, A_{Sy}, A_{NC}, A_{ISE}, $A_{Sy}D_D$, $A_{Sy}D_{nil}$, $A_{NC}D_D$, $A_{NC}D_{nil}$, $A_{ISE}D_D$, $A_{ISE}D_{nil}$, $A_{Sy}D_DA_E$, $A_{Sy}D_DA_{nil}$, $A_{Sy}D_{nil}A_E$, $A_{Sy}D_{nil}A_{DI}$, $A_{Sy}D_DA_E$, $A_{Sy}D_DA_{DI}$, $A_{Sy}D_DA_{nil}$, $A_{NC}D_{nil}A_E$, $A_{NC}D_{nil}A_{DI}$, $A_{ISE}D_DA_{RH}$, and $A_{ISE}D_{ND}A_{RH}$. The function $\longrightarrow$ defines the game transitions such that (W x Σ) $\rightarrow$ W. The possible transitions are:

- start x Sy $\rightarrow$ A_{Sy} where Sy $\in \Sigma_A$

- start x NC $\rightarrow$ A_{NC} where NC $\in \Sigma_A$

- start x ISE $\rightarrow A_{ISE}$ where ISE $\in \Sigma_A$

- A_{Sy} x D $\rightarrow A_{Sy}D_D$ where D $\in \Sigma_D$

- A_{Sy} x nil $\rightarrow A_{Sy}D_{nil}$ where nil $\in \Sigma_D$

- A_{NC} x D $\rightarrow A_{NC}D_D$ where D $\in \Sigma_D$

- A_{NC} x nil $\rightarrow A_{NC}D_{nil}$ where nil $\in \Sigma_D$

- A_{ISE} x D $\rightarrow A_{ISE}D_D$ where D $\in \Sigma_D$

- A_{ISE} x nil $\rightarrow A_{ISE}D_{nil}$ where nil $\in \Sigma_D$

- $A_{Sy}D_D$ x E $\rightarrow A_{Sy}D_D A_E$ where E $\in \Sigma_A$

- $A_{Sy}D_D$ x nil $\rightarrow A_{Sy}D_D A_{nil}$ where nil $\in \Sigma_A$

- $A_{Sy}D_{nil}$ x E $\rightarrow A_{Sy}D_{nil}A_E$ where E $\in \Sigma_A$

- $A_{Sy}D_{nil}$ x DI $\rightarrow A_{Sy}D_{nil}A_{DI}$ where DI $\in \Sigma_A$

- $A_{NC}D_D$ x E $\rightarrow A_{Sy}D_D A_E$ where E $\in \Sigma_A$

- $A_{NC}D_D$ x DI $\rightarrow A_{Sy}D_D A_{DI}$ where DI $\in \Sigma_A$

- $A_{NC}D_D$ x nil $\rightarrow A_{Sy}D_D A_{nil}$ where nil $\in \Sigma_A$

- $A_{NC}D_{nil}$ x E $\rightarrow A_{NC}D_{nil}A_E$ where E $\in \Sigma_A$

- $A_{NC}D_{nil}$ x DI $\rightarrow A_{NC}D_{nil}A_{DI}$ where DI $\in \Sigma_A$

- $A_{ISE}D_D$ x RH $\rightarrow A_{ISE}D_D A_{RH}$ where RH $\in \Sigma_A$

- $A_{ISE}D_{nil}$ x RH $\rightarrow A_{ISE}D_{nil}A_{RH}$ where RH $\in \Sigma_A$

The initial state of the game w_0, is equal to start $\in$ W, where $\overrightarrow{start} = \{A_{Sy}, A_{NC}, A_{ISE}\}$. The belief context β_0 is set to 0.8 (which is randomly chosen for this example), the update function for the belief context $\hookrightarrow$ simply makes no modification to the belief contexts (i.e. $\beta 0 \hookrightarrow \beta_{new}$ implies $\beta 0 = \beta_{new}$). The CMS $\mathcal{C}$, and the initial value $\mathcal{C}_0$ is set to the NEMS value for the hypergame. The fear-of-being-outguessed Υ, is set to an initial value γ of zero. The update function $\twoheadrightarrow$ maps $\gamma \twoheadrightarrow \Upsilon = 0$. Therefore there is no update. The χ function assigns the player whose turn it is to the game state w $\in$ W where W $\rightarrow$ N. For example, $\chi(start) = $ A, while $\chi(A_C) = \chi(A_D) = $ D.

7.5.2.3 Game Mapping.

For Step 3: Game Mapping, an extensive form game tree $\mathcal{T}$ is associated with the SCADA security game arena, ϕ_{SCADA}^{T}. The extensive form game tree is defined as:

$$\mathcal{T} = (S, \Rightarrow, s_0, \lambda)$$

where (S, $\Rightarrow$) is a countably infinite tree rooted at s_0 with edges from Σ. The nodes of the tree are given by S, where S = {root, A_{Sy}, A_{NC}, $A_{Sy}D_D$, $A_{Sy}D_{nil}$, $A_{NC}D_D$, $A_{NC}D_{nil}$, $A_{Sy}D_DA_E$, $A_{Sy}D_DA_{nil}$, $A_{Sy}D_{nil}A_E$, $A_{Sy}D_{nil}A_{DI}$, $A_{Sy}D_DA_E$, $A_{Sy}D_DA_{DI}$, $A_{Sy}D_DA_{nil}$, $A_{NC}D_{nil}A_E$, and $A_{NC}D_{nil}A_{DI}$}. The root of the tree denoted s_0, is equal to root $\in$ S. The $\overset{x}{\Rightarrow}$ is the function that moves between nodes of the tree using the edge denoted by x $\in \Sigma$.

The possible nodes transitions are {root $\overset{Sybil}{\Longrightarrow} A_{Sy}$, root $\overset{NodeCompromise}{\Longrightarrow} A_{NC}$, root $\overset{ISE}{\Longrightarrow} A_{ISE}$ where Sybil, Node Compromise, ISE $\in \sigma_A$} $\cup$ {$A_{Sy} \overset{D}{\Rightarrow} A_{Sy}D_D$, $A_{Sy} \overset{nil}{\Rightarrow} A_{Sy}D_{nil}$, $A_{NC} \overset{D}{\Rightarrow} A_{NC}D_D$, $A_{NC} \overset{nil}{\Rightarrow} A_{NC}D_{nil}$, $A_{ISE} \overset{nil}{\Rightarrow} A_{ISE}D_{nil}$, $A_{ISE} \overset{D}{\Rightarrow} A_{ISE}D_D$ where D, nil $\in \sigma_D$} $\cup$ {$A_{Sy}D_D \overset{E}{\Rightarrow} A_{Sy}D_DA_E$, $A_{Sy}D_D \overset{nil}{\Rightarrow} A_{Sy}D_DA_{nil}$, $A_{Sy}D_{nil} \overset{E}{\Rightarrow} A_{Sy}D_{nil}A_E$, $A_{Sy}D_{nil} \overset{DI}{\Rightarrow} A_{Sy}D_{nil}A_{DI}$, $A_{NC}D_D \overset{E}{\Rightarrow} A_{NC}D_DA_E$, $A_{NC}D_D \overset{DI}{\Rightarrow} A_{NC}D_DA_{DI}$, $A_{NC}D_D \overset{nil}{\Rightarrow} A_{NC}D_DA_{nil}$, $A_{NC}D_{nil} \overset{E}{\Rightarrow} A_{NC}D_{nil}A_E$, $A_{NC}D_{nil} \overset{DI}{\Rightarrow}$

$A_{NC}D_{nil}A_{DI}$, $A_{ISE}D_D \overset{RH}{\Longrightarrow} A_{ISE}D_DA_{RH}$, $A_{ISE}D_{nil} \overset{RH}{\Longrightarrow} A_{ISE}D_{nil}A_{RH}$ where E, DI, nil, RH $\in \sigma_A\}$.

The function λ is S $\to$ W, where

- $\lambda(s_0) = w_0$

- $\forall s, s' \in S$, if s $\overset{a}{\Rightarrow}$ s' then $\lambda(s) \overset{a}{\Rightarrow} \lambda(s')$

- if $\lambda(s)$ = w and w $\overset{a}{\to}$ w' there exists $s' \in$ S s.t. s $\overset{a}{\Rightarrow}$ s' and $\lambda(s') = w'$

7.5.2.4   Path Structuring.

For Step 4: Path Structuring, the players individual paths through the game tree are defined. Theses paths are later used to define the strategies, but are not required. Defining the paths reduces the amount of notation required for the strategy definitions. The model for the SCADA security temporal hypergame can be represented by $M_{SCADA} = (\mathcal{T}, \mathcal{V})$. The game tree, $\mathcal{T}$ is given from the previous game mapping and the valuation function $\mathcal{V}$ is given by:

- $\mathcal{V}(p_{int}) = \{s_0\}$

- $\mathcal{V}(p_{domD}) = \{A_{Sy}, A_{Sy}D_DA_E\}$

- $\mathcal{V}(p_{worst_A}) = \{A_{NC}, A_{NC}D_D\}$

- $\mathcal{V}(p_{worst_D}) = \{A_{NC}, A_{NC}D_{nil}\}$ or $\{A_{Sy}, A_{Sy}D_{nil}\}$

- $\mathcal{V}(p_{expA}) = \{A_{Sy}, A_{Sy}D_DA_{DI}\}$

When the hypergame is considered, the additional valuation function is included:

- $\mathcal{V}(p_{domA}) = \{A_{ISE}, A_{ISE}D_D\}$ or $\{A_{ISE}, A_{ISE}D_{nil}\}$

7.5.2.5 Define Player Strategies.

For Step 5: Define Player Strategies, strategies for each player are defined in terms of the states and transitions, as well as the paths defined previously. In this game the Attacker's strategy can be defined as:

$$Strat^A \equiv ([p_{int} \mapsto Sy]^A) \cdot ([p_{int} \mapsto NC]^A) \cdot ([p_{int} \mapsto ISE]^A) \cdot ([p_{domD} \mapsto D]^D \Rightarrow$$

$$[p_{max_A} \mapsto E]^A) \cdot ([p_{domD} \mapsto nil]^D \Rightarrow [p_{max_A} \mapsto DI]^A) \cdot ([p_{domD} \mapsto D]^D \Rightarrow$$

$$[p_{domA} \mapsto RH]^A) \cdot ([p_{domD} \mapsto nil]^D \Rightarrow [p_{domA} \mapsto RH]^A) \cdot ([\; \Box \; [turn_A \mapsto Sy]^A \Rightarrow$$

$$[turn_A \mapsto E]^A \vee [turn_A \mapsto DI]^A) ([\; \Box \; [turn_A \mapsto NC]^A \Rightarrow [turn_A \mapsto E]^A \vee$$

$$[turn_A \mapsto DI]^A) \cdot ([\; \Box \; [turn_A \mapsto ISE]^A \Rightarrow [turn_A \mapsto RH]^A)$$

The Defender's strategy can be defined as:

$$Strat^D \equiv ([p_{int} \mapsto Sy]^A \Rightarrow [p_{domD} \mapsto D]^D) \cdot ([p_{int} \mapsto NC]^A \Rightarrow [p_{domD} \mapsto D]^D) \cdot$$

$$([p_{int} \mapsto Sy]^A \Rightarrow [p_{worst_D} \mapsto nil]^D) \cdot ([p_{int} \mapsto NC]^A \Rightarrow [p_{worst_D} \mapsto nil]^D)$$

7.5.2.6 Analyze Model.

For the final Step 6: Analyze Model, the previous definitions are used to logical analysis the constructed model. From the previous definitions, $Strat^D \leadsto_D (p_{domD} \vee p_{worst_D})$, which means the defender can either follow the dominate strategy or the worst case strategy. If the defender plays "Defend" then the defender can ensure the worst outcome is avoided:

$$[turn_D \mapsto D]^D \leadsto_D \neg p_{worst_D}$$

This indicates the standard result of the game where the defender always desires to defend. With this, the defender would expect the attacker to reach the expected best case p_{expA}. Which means the defender expects $Strat^A \leadsto_A p_{expA}$. Since there is a hypergame, the strategy is really $Strat^A \leadsto_A p_{domA}$.

The attacker follows $\neg\text{ISE}?[turn_A \mapsto ISE]^A \land \text{ISE}?[turn_A \mapsto ISE]^A$. This allows the attacker to achieve the best outcome, while allowing the defender to follow their perceived dominant strategy.

For the defender the strategy is to always defend:

$$[turn_D \mapsto D]^D \leadsto_D \neg p_{worst_D}$$

For the attacker the strategy is to play the hidden hypergame strategy of always choosing the ISE in the first round and then the RH action in the second round.

$$\neg\text{ISE}?[turn_A \mapsto ISE]^A \land \text{ISE}?[turn_A \mapsto ISE]^A \leadsto_A p_{expA}$$

7.5.3 Cyber Physical SCADA Security Temporal Hypergame Example - Defender View.

This section extends the previous simple SCADA security example with additional actions that can lead to misperceptions, in order to validate more of the temporal hypergame framework. This example is designed to show how misperceptions can lead to different results based on the view of the defender. The payoff function is denoted $U_d(p, a)$, of player p at a decision node of depth d as a result of action a. Action a can be either taken by player p or the opponent. The payoff function (utility) is calculated by:

$$U_d(p, a) = U_{d-1}(p, a') + B(p, a, d)$$

$B(p, a, d)$ denotes the behavior of the impact of the action and a' denotes the action of p's opponent. At the root of the game tree (i.e. start of the game) there are no previous payoffs, so the initial payoff is $U_0(p, nil) = (0, 0)$ where nil is no action. $B(p, a, d)$ is the behavior of the impact of the action, as shown in Table 7.5. It depends on the action a, the player p, and the depth d of the game tree (shown in Figure 7.10) as an indicator of the game's advancement.

189

Table 7.5: SCADA Behavior of the Impact.

$B(p, a, d)$	a is A's action	a is D's action
p = A (Attacker)	$d * Impact(a)$	0
p = D (Defender)	$-Impact(a)^d$	$Impact(a)$

The impact function shown in Table 7.5, determines the impact of confidentiality, integrity, and availability. Let C(a), I(a), and A(a) be the confidentiality, integrity, and availability with the corresponding weights w_C, w_I, and w_A of action a. The function is then defined as:

$$Impact(a) = w_C C(a) + w_I I(a) + w_A A(a)$$

The weight sum to 1 and are fixed at $w_C = 0.1$, $w_I = 0.6$, and $w_A = 0.3$. The impact function is calculated according to Table 7.6. Not defend and not attack have no impact on the model and are excluded from the table. In Table 7.6, the value 1 is considered low, 4 is moderate, and 8 is high, in terms of impact. The SCADA impact function is expanded by adding four new actions - virus, ruin, hide, and disconnect.

The virus action is only available to the attacker in the first round (the attacker's initial decision). The ruin or hide action is only available to the attacker if the attacker chose infect support equipment in the first round. The disconnect action is only available to the defender in the first round, after that it is not available. The virus action represents a cyber weapon that may be used in a future conflict. The defenders option to disconnect represents a third world country, such as one in Africa, that may have limited connectivity to the outside world and therefore have limited infection vectors.

Table 7.6: SCADA Model Impact Function.

Description	C(a)	I(a)	A(a)	Impact(a)
Defend	6	6	1	4.5
Disconnect	8	8	8	8.0
Sybil	6	1	1	1.5
Node Compromise	6	1	1	1.5
Eavesdropping	8	1	1	1.7
Data Injection	1	8	6	6.7
Virus	6	3	3	3.3
Ruin	2	8	8	7.4
Hide	4	0	0	0.4

The extended SCADA sensor nework game is shown in Figure 7.10. This game tree shows the addition of the additional actions for the attacker and defender in their respective rounds of play.

7.5.3.1 Initial Game Definition.

To apply Step 1: Initial Game Definition of the temporal hypergame framework to the information from the SCADA model, it is necessary to create a temporal hypergame model. Let the temporal hypergame be:

$$H^T_{SCADA} \triangleq \{G^T_D\}$$

The perceived game G^T_{SCADA}:

$$G^T_{SCADA} \triangleq (N, \Sigma_{Full}, \Phi^T, \{\preceq^i\}_{i \in N})$$

Where N = {Attacker (A), Defender (D)} is the set of players. Σ_{Full} is the finite set of player actions such that $\Sigma_{Full} = \Sigma_A \cup \Sigma_D$, $\Sigma_A = \{Sybil(Sy), NodeCompromise(NC),$

$Eavesdropping(E), DataInjection(DI), NotAttack(nil), Virus(V), Ruin(R), Hide(H)\}$,
and $\Sigma_D = \{Defend(D), NotDefend(nil), Diconnect(DC)\}$. In this model the action virus is known to the defender, but the action disconnect is not known to the attacker.

The preference ordering function for the i_{th} player $\preceq^i = u_i(x,y) \le u_i(x',y)$ for i $\in$ N. This means for attacker $\preceq^A$ indicates the outcome (x,y) is preferred to outcome (x',y) if x $\le x'$ $\forall (x,y) \in W$. For defender $\preceq^D$ indicates the outcome (x,y) is preferred to outcome (x,y') if y $\le y'$ $\forall (x,y) \in W$. Notice the Nash equilibrium concept is defined in the preference relation, $\preceq^i$, assuming both players are rational.

The variable payoff function is denoted by:

$$\mathcal{V} \triangleq \{\omega, \omega_{init}, \psi, \delta\}$$

The payoff function variables are given by $\omega = \{w_C, w_I, w_A, d, Impact(a)\}$. The initial values ω_{init}, are set to 0 except for $w_C = 0.1$, $w_I = 0.6$, and $w_A = 0.3$. The function ψ works as follows:

- If variable Impact(a)

    - If a = Defend, then Impact = 4.5

    - If a = Disconnect, then Impact = 8.0

    - if a = Sybil, then Impact = 1.5

    - if a = Node Compromise, then Impact = 1.5

    - if a = Eavesdropping, then Impact = 1.7

    - if a = Data Injection, then Impact = 6.7

    - if a = Infect Support Equipment, then Impact = 3.3

    - if a = Ruin, then Impact = 4.1

    - if a = Hide, then Impact = 0.4

- if variable d, then d = d + 1

With the proposed model there are no constraints to be applied.

7.5.3.2 *Identification of States and Transitions.*

For Step 2: Identification of States and Transitions of the temporal hypergame framework, it is necessary to define the rules of the game the players must follow and is derived from the game tree. The game arena is given by:

$$\Phi^T_{SCADA} \triangleq (W, \longrightarrow, w_0, \beta, \mathcal{C}, \hookrightarrow, \beta_0, \mathcal{C}_0, \Upsilon, \twoheadrightarrow, \gamma, \mathcal{V}, \chi)$$

W consists of twenty-nine game states, including start, A_{Sy}, A_{NC}, A_V, $A_{Sy}D_D$, $A_{Sy}D_{nil}$, $A_{Sy}D_{DC}$, $A_{NC}D_D$, $A_{NC}D_{nil}$, $A_{NC}D_{DC}$, $A_V D_D$, $A_V D_{nil}$, $A_V D_{DC}$, $A_{Sy}D_D A_{DI}$, $A_{Sy}D_D A_E$, $A_{Sy}D_{nil}A_{DI}$, $A_{Sy}D_{nil}A_E$, $A_{Sy}D_{DC}A_{DI}$, $A_{Sy}D_{DC}A_E$, $A_{NC}D_{nil}A_{DI}$, $A_{NC}D_{nil}A_E$, $A_{NC}D_{DC}A_{DI}$, $A_{NC}D_{DC}A_E$, $A_V D_D A_R$, $A_V D_D A_H$, $A_V D_{nil}A_R$, $A_V D_{nil}A_H$, $A_V D_{DC}A_R$, $A_V D_{DC}A_H$. The function $\longrightarrow$ defines the game transitions such that (W x Σ) $\rightarrow$ W. The possible transitions are:

- start x Sy $\rightarrow A_{Sy}$ where Sy $\in \Sigma_A$

- start x NC $\rightarrow A_{NC}$ where NC $\in \Sigma_A$

- start x V $\rightarrow A_V$ where V $\in \Sigma_A$

- A_{Sy} x D $\rightarrow A_{Sy}D_D$ where D $\in \Sigma_D$

- A_{Sy} x nil $\rightarrow A_{Sy}D_{nil}$ where nil $\in \Sigma_D$

- A_{Sy} x DC $\rightarrow A_{Sy}D_{DC}$ where DC $\in \Sigma_D$

- A_{NC} x D $\rightarrow A_{NC}D_D$ where D $\in \Sigma_D$

- A_{NC} x nil $\rightarrow A_{NC}D_{nil}$ where nil $\in \Sigma_D$

- A_{NC} x DC $\rightarrow A_{NC}D_{DC}$ where DC $\in \Sigma_D$

- A_V x D $\rightarrow A_V D_D$ where D $\in \Sigma_D$

- A_V x nil $\rightarrow A_V D_{nil}$ where nil $\in \Sigma_D$

- A_V x DC $\rightarrow A_V D_{DC}$ where DC $\in \Sigma_D$

- $A_{Sy}D_D$ x E $\rightarrow A_{Sy}D_D A_E$ where E $\in \Sigma_A$

- $A_{Sy}D_D$ x DI $\rightarrow A_{Sy}D_D A_{DI}$ where DI $\in \Sigma_A$

- $A_{Sy}D_{nil}$ x E $\rightarrow A_{Sy}D_{nil}A_E$ where E $\in \Sigma_A$

- $A_{Sy}D_{nil}$ x DI $\rightarrow A_{Sy}D_{nil}A_{DI}$ where DI $\in \Sigma_A$

- $A_{Sy}D_{DC}$ x E $\rightarrow A_{Sy}D_{DC}A_E$ where E $\in \Sigma_A$

- $A_{Sy}D_{DC}$ x DI $\rightarrow A_{Sy}D_{DC}A_{DI}$ where DI $\in \Sigma_A$

- $A_{NC}D_D$ x E $\rightarrow A_{Sy}D_D A_E$ where E $\in \Sigma_A$

- $A_{NC}D_D$ x DI $\rightarrow A_{Sy}D_D A_{DI}$ where DI $\in \Sigma_A$

- $A_{NC}D_{nil}$ x E $\rightarrow A_{NC}D_{nil}A_E$ where E $\in \Sigma_A$

- $A_{NC}D_{nil}$ x DI $\rightarrow A_{NC}D_{nil}A_{DI}$ where DI $\in \Sigma_A$

- $A_{NC}D_{DC}$ x E $\rightarrow A_{NC}D_{DC}A_E$ where E $\in \Sigma_A$

- $A_{NC}D_{DC}$ x DI $\rightarrow A_{NC}D_{DC}A_{DI}$ where DI $\in \Sigma_A$

- $A_V D_D$ x R $\rightarrow A_V D_D A_R$ where R $\in \Sigma_A$

- $A_V D_D$ x H $\rightarrow A_V D_D A_H$ where H $\in \Sigma_A$

- $A_V D_{nil}$ x R $\rightarrow A_V D_{nil}A_R$ where R $\in \Sigma_A$

- $A_V D_{nil}$ x H $\rightarrow A_V D_{nil} A_H$ where H $\in \Sigma_A$

- $A_V D_{DC}$ x R $\rightarrow A_V D_{DC} A_R$ where R $\in \Sigma_A$

- $A_V D_{DC}$ x H $\rightarrow A_V D_{DC} A_H$ where H $\in \Sigma_A$

The initial state of the game w_0, is equal to start $\in$ W, where $\overrightarrow{start} =$ $\{A_{Sy}, A_{NC}, A_V\}$. The belief context β_0 is set to 0.8 (which is selected randomly for this example), the update function for the belief context $\hookrightarrow$ simply makes no modification to the belief contexts (i.e. $\beta 0 \hookrightarrow \beta_{new}$ implies $\beta 0 = \beta_{new}$). The CMS $\mathcal{C}$, and the initial value $\mathcal{C}_0$ is set to the NEMS value for the hypergame. The fear-of-being-outguessed Υ, is set to an initial value γ of zero. The update function $\twoheadrightarrow$ maps $\gamma \twoheadrightarrow \Upsilon = 0$. Therefore there is no update. The χ function assigns he player whose turn it is to the game state w $\in$ W where W $\rightarrow$ N. For example, $\chi(start) =$ A, while $\chi(A_C) = \chi(A_D) =$ D.

7.5.3.3 Game Mapping.

For Step 3: Game Mapping, an extensive form game tree $\mathcal{T}$ is associated with the SCADA security game arena, ϕ^T_{SCADA}. The extensive form game tree is defined as:

$$\mathcal{T} = (S, \Rightarrow, s_0, \lambda) \tag{7.6}$$

where (S, $\Rightarrow$) is a countably infinite tree rooted at s_0 with edges from Σ. The nodes of the tree are given by S, where S = {root, A_{Sy}, A_{NC}, A_V, $A_{Sy}D_D$, $A_{Sy}D_{nil}$, $A_{Sy}D_{DC}$, $A_{NC}D_D$, $A_{NC}D_{nil}$, $A_{NC}D_{DC}$, $A_V D_D$, $A_V D_{nil}$, $A_V D_{DC}$, $A_{Sy}D_D A_E$, $A_{Sy}D_D A_{DI}$, $A_{Sy}D_{nil}A_E$, $A_{Sy}D_{nil}A_{DI}$, $A_{Sy}D_{DC}A_E$, $A_{Sy}D_{DC}A_{DI}$, $A_{NC}D_{nil}A_E$, $A_{NC}D_{nil}A_{DI}$, $A_{NC}D_{DC}A_E$, $A_{NC}D_{DC}A_{DI}$, $A_V D_D A_R$, $A_V D_D A_H$, $A_V D_{nil}A_R$, $A_V D_{nil}A_H$, $A_V D_{DC}A_R$, $A_V D_{DC}A_H$}. The root of the tree denoted s_0, is equal to root $\in$ S. The $\overset{x}{\Rightarrow}$ is the function that moves between nodes of the tree using the edge denoted by x $\in \Sigma$.

The possible nodes transitions are $\{$root $\xRightarrow{Sybil} A_{Sy}$, root $\xRightarrow{NodeCompromV} A_{NC}$, root $\xRightarrow{V} A_V$ where Sy, NC, V $\in \sigma_A\} \cup \{A_{Sy} \xRightarrow{D} A_{Sy}D_D$, $A_{Sy} \xRightarrow{nil} A_{Sy}D_{nil}$, $A_{Sy} \xRightarrow{DC} A_{Sy}D_{DC}$, $A_{NC} \xRightarrow{D} A_{NC}D_D$, $A_{NC} \xRightarrow{nil} A_{NC}D_{nil}$, $A_{NC} \xRightarrow{DC} A_{NC}D_{DC}$, $A_V \xRightarrow{nil} A_V D_{nil}$, $A_V \xRightarrow{D} A_V D_D$, $A_V \xRightarrow{DC} A_V D_{DC}$ where D, nil, DC $\in \sigma_D\} \cup \{A_{Sy}D_D \xRightarrow{E} A_{Sy}D_D A_E$, $A_{Sy}D_D \xRightarrow{nil} A_{Sy}D_D A_{nil}$, $A_{Sy}D_{nil} \xRightarrow{E} A_{Sy}D_{nil}A_E$, $A_{Sy}D_{nil} \xRightarrow{DI} A_{Sy}D_{nil}A_{DI}$, $A_{Sy}D_{DC} \xRightarrow{E} A_{Sy}D_{DC}A_E$, $A_{Sy}D_{DC} \xRightarrow{DI} A_{Sy}D_{DC}A_{DI}$, $A_{NC}D_D \xRightarrow{E} A_{NC}D_D A_E$, $A_{NC}D_D \xRightarrow{DI} A_{NC}D_D A_{DI}$, $A_{NC}D_D \xRightarrow{nil} A_{NC}D_D A_{nil}$, $A_{NC}D_{nil} \xRightarrow{E} A_{NC}D_{nil}A_E$, $A_{NC}D_{nil} \xRightarrow{DI} A_{NC}D_{nil}A_{DI}$, $A_{NC}D_{DC} \xRightarrow{E} A_{NC}D_{DC}A_E$, $A_{NC}D_{DC} \xRightarrow{DI} A_{NC}D_{DC}A_{DI}$, $A_V D_D \xRightarrow{R} A_V D_D A_R$, $A_V D_D \xRightarrow{H} A_V D_D A_H$, $A_V D_{nil} \xRightarrow{R} A_V D_{nil}A_R$, $A_V D_{nil} \xRightarrow{H} A_V D_{nil}A_H$ $A_V D_{DC} \xRightarrow{R} A_V D_{DC}A_R$, $A_V D_{DC} \xRightarrow{H} A_V D_{DC}A_H$, where E, DI, nil, R, H $\in \sigma_A\}$.

The function λ is S $\rightarrow$ W, where

- $\lambda(s_0) = w_0$

- $\forall s, s' \in S$, if s $\xRightarrow{a} s'$ then $\lambda(s) \xRightarrow{a} \lambda(s')$

- if $\lambda(s) =$ w and w $\xrightarrow{a} w'$ there exists $s' \in$ S s.t. s $\xRightarrow{a} s'$ and $\lambda(s') = w'$

7.5.3.4   Path Structuring.

For Step 4: Path Structuring, the players individual paths through the game tree are defined. Theses paths are later used to define the strategies, but are not required. Defining the paths reduces the amount of notation required for the strategy definitions. The model for the extended SCADA security temporal hypergame can be represented by $M_{SCADA} = (\mathcal{T}, \mathcal{V})$. The game tree, $\mathcal{T}$ is given from the previous game mapping and the valuation function $\sqsubseteq$ is given by:

- $\mathcal{V}(p_{int}) = \{s_0\}$

- $\mathcal{V}(p_{domA}) = \{A_V, A_V D_D, A_V D_D A_R\}$

- $\mathcal{V}(p_{domD_{exp}}) = \{A_{Sy}, A_{Sy}D_D\}$ or $\{A_{NC}, A_{NC}D_D\}$ or $\{A_V, A_V D_D\}$

196

- $\mathcal{V}(p_{worst_A}) = \{A_{NC},\ A_{NC}D_D\}$

- $\mathcal{V}(p_{worst_D}) = \{A_{NC},\ A_{NC}D_{nil}\}$ or $\{A_{Sy},\ A_{Sy}D_{nil}\}$ or $\{A_V,\ A_V D_{nil}\}$

When the hypergame advantage (defender can Disconnect) is considered, the additional functions are included; while the other functions remain the same:

- $\mathcal{V}(p_{domD_{hyp}}) = \{A_{Sy},\ A_{Sy}D_{DC}\}$ or $\{A_{NC},\ A_{NC}D_{DC}\}$ or $\{A_{NC},\ A_{NC}D_D\}$ or $\{A_V,\ A_V D_{DC}\}$

- $\mathcal{V}(p_{worstA_{hyp}}) = \{A_V,\ A_V D_{DC},\ A_V D_{DC}A_R\}$ or $\{A_V,\ A_V D_{DC},\ A_V D_{DC}A_H\}$ or $\{A_{Sy},\ A_{Sy}D_{DC},\ A_{Sy}D_{DC}A_{DI}\}$ or $\{A_{Sy},\ A_{Sy}D_{NC}D_{DC}A_E\}$

The $p_{domD_{hyp}}$ represents the dominant path for the defender from the attacker's point-of-view. The attacker has failed to account for the defender's disconnect option. While the $p_{domD_{exp}}$ represents the dominant path for the defender when the defender's advantage (Disconnect) is considered. The attacker is unaware of this option, and therefore assumes $p_{domD_{exp}}$ is the expected outcome for a rational defender.

7.5.3.5 Define Player Strategies.

For Step 5: Define Player Strategies, strategies for each player are defined in terms of the states and transitions, as well as the paths defined previously. In this game the attacker's strategy can be defined as:

$$Strat^A \equiv ([p_{int} \mapsto Sy]^A) \cdot ([p_{int} \mapsto NC]^A) \cdot ([p_{int} \mapsto V]^A) \cdot ([\ \square\ [turn_A \mapsto Sy]^A \Rightarrow$$
$$[turn_A \mapsto E]^A \vee [turn_A \mapsto DI]^A\}) \cdot ([\ \square\ [turn_A \mapsto NC]^A \Rightarrow [turn_A \mapsto E]^A \vee$$
$$[turn_A \mapsto DI]^A \vee [turn_A \mapsto nil]^A\}) \cdot ([\ \square\ [turn_A \mapsto V]^A \Rightarrow [turn_A \mapsto R]^A \vee$$
$$[turn_A \mapsto H]^A\}) \cdot ([p_{domD_{exp}} \mapsto D]^D \Rightarrow [p_{max_A} \mapsto R]^A) \cdot ([p_{domD_{exp}} \mapsto nil]^D \Rightarrow$$
$$[p_{max_A} \mapsto H]^A)$$

From the attacker's perspective, the defender's strategy is defined as:

$$Strat_A^D \equiv ([p_{int} \mapsto Sy]^A \Rightarrow [p_{domD_{exp}} \mapsto D]^D) \cdot ([p_{int} \mapsto NC]^A \Rightarrow [p_{domD_{exp}} \mapsto D]^D) \cdot$$

$$([p_{int} \mapsto V]^A \Rightarrow [p_{domD_{exp}} \mapsto D]^D) \cdot ([p_{int} \mapsto Sy]^A \Rightarrow [p_{worst_D} \mapsto nil]^D) \cdot$$

$$([p_{int} \mapsto NC]^A \Rightarrow [p_{worst_D} \mapsto nil]^D) \cdot ([p_{int} \mapsto V]^A \Rightarrow [p_{worst_D} \mapsto nil]^D)$$

But the defender has an extra action available (Disconnect) that the attacker is not aware of, so the defender's strategy can be defined as:

$$Strat^D \equiv ([p_{int} \mapsto Sy]^A \Rightarrow [p_{domD_{hyp}} \mapsto DC]^D) \cdot ([p_{int} \mapsto NC]^A \Rightarrow$$

$$[p_{domD_{hyp}} \mapsto DC]^D \vee [p_{domD_{hyp}} \mapsto D]^D) \cdot ([p_{int} \mapsto V]^A \Rightarrow [p_{domD_{hyp}} \mapsto DC]^D) \cdot$$

$$([p_{int} \mapsto Sy]^A \Rightarrow [p_{worst_D} \mapsto nil]^D) \cdot ([p_{int} \mapsto NC]^A \Rightarrow [p_{worst_D} \mapsto nil]^D) \cdot$$

$$([p_{int} \mapsto V]^A \Rightarrow [p_{worst_D} \mapsto nil]^D)$$

7.5.3.6 *Analyze Model.*

For the final Step 6: Analyze Model, the previous definitions are used to logically analyze the constructed model. From the previous definitions, $Strat_A^D \leadsto_D (p_{domD_{exp}} \vee p_{worst_D})$, which means the defender can either follow the dominant strategy or the worst case strategy from the view of the attacker. A rational defender is always assumed to follow the dominant strategy. In this case if the defender plays Defend then the defender can ensure the worst outcome is avoided:

$$[turn_D \mapsto D]^D \leadsto_D \neg p_{worst_D}$$

This represents the standard result of the game where the defender always desires to defend. With this, the defender would expect the attacker to reach the expected best case p_{exp_A}. Which means the defender would expect $Strat^A \leadsto_A p_{domA}$.

The attacker follows $\neg V?[turn_A \mapsto V]^A \wedge V?[turn_A \mapsto V]^A$. This allows the attacker to achieve the best outcome, while allowing the defender to follow their perceived dominant strategy.

For the defender the solution is always to use the "Defend" strategy:

$$[turn_D \mapsto D]^D \leadsto_D \neg p_{worst_D}$$

The attacker's strategy is to always choose the Virus (V) in the first round and then the Ruin (R) action in the third round. This strategy represents the temporal relationship between the actions for the attacker. If the attacker chooses any other available action in the first round, then the Ruin action is not available in the next round.

$$\neg V?[turn_A \mapsto V]^A \wedge V?[turn_A \mapsto V]^A \leadsto_A p_{domA}$$

Up to this point the analysis has focused on the solution to the game without considering the hypergame advantage for the defender. From the previous definitions, $Strat^D \leadsto_D (p_{domD_{hyp}} \vee p_{worst_D})$, which means the defender can either follow the hypergame strategy or the worst case strategy. It is assumed that a rational defender always plays to the hypergame strategy, since it is dominant. In this case, if the defender plays "Defend" or the hidden hypergame strategy "Disconnect", then the defender can ensure the worst outcome is avoided:

$$[turn_D \mapsto D]^D \vee [turn_D \mapsto DC]^D \leadsto_D \neg p_{worst_D}$$

This shows the hypergame result of the game where the defender would expect the attacker to reach the expected worst case p_{worst_D}. Which means the defender expects $Strat^A \leadsto_A p_{worstA_{hyp}}$.

The attacker still follows $\neg V?[turn_A \mapsto V]^A \wedge V?[turn_A \mapsto V]^A$. This allows the attacker to achieve their perceived best outcome in the temporal sense, which is $Strat^A \leadsto_A p_{domA}$ from the previous analysis. Because of the hypergame, the attacker really ends up in the worst outcome $Strat^A \leadsto_A p_{worstA_{hyp}}$.

Since the solution to the hypergame is for the defender to play Defend or Disconnect and the attacker to play Virus followed by Ruin, the defender can eliminate

Defend. By playing only Disconnect, the defender can guarantee the best possible outcome.

$$[turn_D \mapsto DC]^D \leadsto_D \neg p_{worst_D}$$

The attacker's strategy is to always play the action Virus (V) in the first round and then Ruin (R) in the third round.

$$[turn_D \mapsto D]^D \vee [turn_D \mapsto DC]^D \leadsto_D \neg p_{worst_D}$$

The temporal hypergame analysis shows that the defender can improve upon their outcome by leveraging the attacker's misperception as the game progresses temporally. It also shows that the defender can guarantee the attacker reaches the worst case outcome instead of the best outcome from the standard game analysis.

7.6 Summary

This chapter applies the temporal hypergame framework presented in the previous chapters to the game theoretic Prisoner's Dilemma, an iterated hypergame, and three cyber physical examples. All of the examples use the six-step process from Section 7.3 for creating a representative model with the temporal hypergame framework. These examples are applied for validation of the temporal hypergame framework in order to show the utility and applicability of the framework. The examples chosen exercise part of the temporal hypergame framework and are not all inclusive. Each example exercises parts of the framework, showing the temporal hypergame framework provides the correct insight into the modeled event.
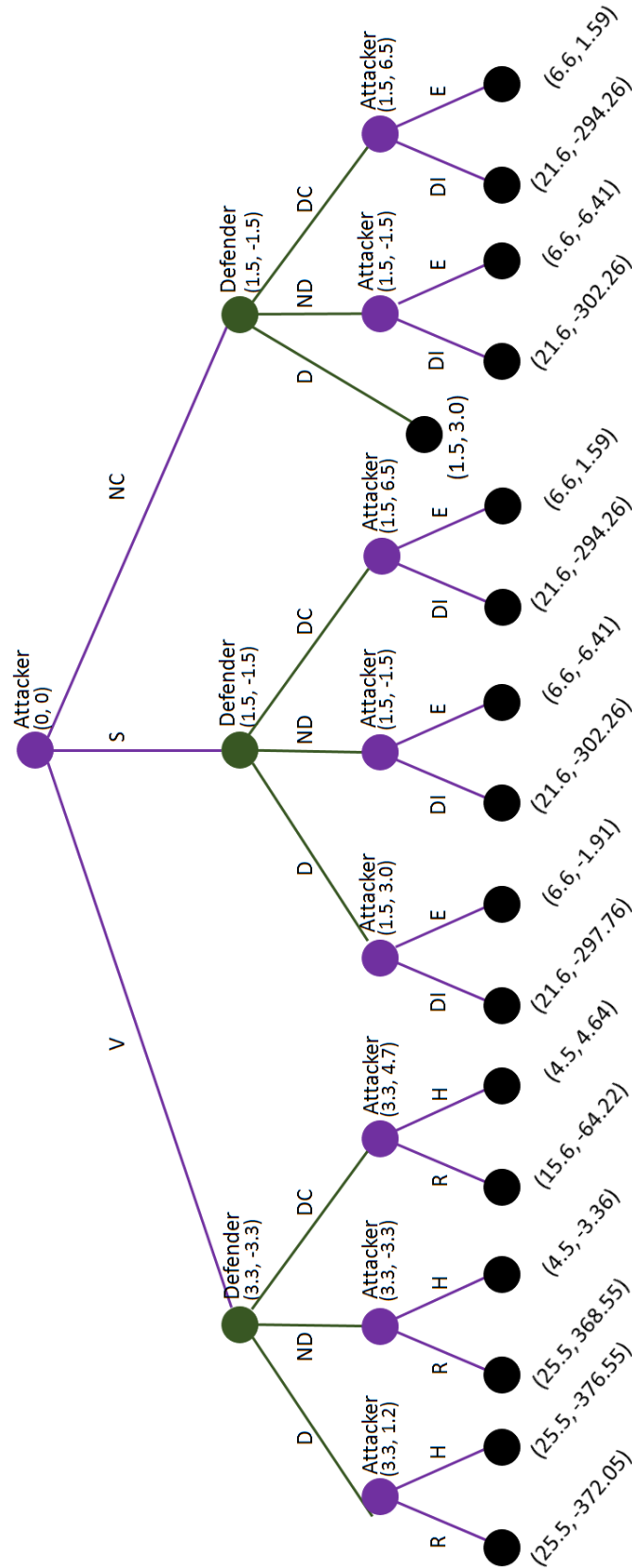
Figure 7.10: Expanded SCADA Sensor Network Game.

VIII.   Conclusions and Future Work

This research presents a temporal hypergame framework to capture the temporal aspects of conflict and decision making. Overall this dissertation presents the first application of temporal logic to hypergames in order to provide a more flexible method for modeling by domain experts. Using this framework the concepts of trust, distrust, and deception are developed and formalized for Hypergame Theory. The framework is applied to a SCADA hypergame, as well as classical game theoretic games, to show that the framework is a realistic modeling method for a variety of applications given its flexibility.

8.1   Conclusions and Findings

Findings 1, 2, 3, and 4 take concepts from Sasaki [310] and generalize the concepts to Vane's hypergame model [356], by defining the base game, difference game, and hyper Nash equilibrium in Chapter 5. The rest of the findings are in Chapter 6. Findings 5, 6, 7, 8, 9, 10, and 11 define trust, misperception and deception over the temporal hypergame model. Findings 13, 14, and 15 relate theorems concerning the SPNE of repeated games to the temporal hypergame framework. The findings are repeated here for reference.

Finding 1. *Let $H = (G^p, G^q)$ be a hypergame with $G_p = (N, \Sigma_p, u_p)$ and $G_q = (N, \Sigma_q, u_q)$ where $p, q \in N$. A normal form game $G = (N, \Sigma, u)$ is called the base game of H iff $u = u_p$, $u = u_q$, and $\Sigma = \Sigma_p \cap \Sigma_q \neq$. Let the base game (BG) of hypergame H be denoted by BG_H.*

Finding 2. *Subgames always have the same base game*

Finding 3. *Let $H = (G^p, G^q)$ be a hypergame with $G_p = (N, \Sigma_p, u_p)$ and $G_q = (N, \Sigma_q, u_q)$ where $p, q \in N$. A normal form game $G = (N, \Sigma, u)$ is called the difference*

game of H iff $u = u_p \cup u_q \setminus u_p \cap u_q$, $\Sigma = \Sigma_p \cup \Sigma_q \setminus \Sigma_p \cap \Sigma_q$, and $u \neq$ or $\Sigma \neq$. Let the difference game of hypergame H be denoted by Δ_H.

Finding 4. *Let $H = (G^p, G^q)$ be a hypergame with $G_p = (N, \Sigma_p, u_p)$ and $G_q = (N, \Sigma_q, u_q)$. Then $a^* \in \Sigma_p \cap \Sigma_q$ is called a stable hyper Nash (SHN) equilibrium iff $a^* \in N(G_p)$ and $a^* \in N(G_q)$ where N(G) represents the Nash equilibriums for game G.*

Finding 5. *Weak Trust* *- Player 1 has weak trust in Player 2 if at a vertex $s' \in S_{\sigma_1}$ $\exists s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

Finding 6. *Strong Trust* *- Player 1 has strong trust in Player 2 if $\forall s' \in S_{\sigma_1}$ $\exists s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

Finding 7. *Weak Distrust* *- Player 1 has weak distrust in Player 2 if at a vertex $s' \in S_{\sigma_1}$ $\nexists s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

Finding 8. *Strong Distrust* *- Player 1 has strong distrust in Player 2 if $\forall s' \in S_{\sigma_1}$ $\nexists s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$.*

Finding 9. *Mistrust* *- Player 1 mistrust Player 2 if $\exists s' \in S_{\sigma_1}$ where Player 1 distrusts Player 2.*

Finding 10. *Misperception* *- Player 2 misperceives the strategy of Player 1 if there is at least one vertex $s' \in S_{\sigma_1}$ $\exists s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s'$ so that when a new strategy σ_* which is equal to σ_1 except that $\nexists s \in S_{\sigma_1}$ and $a \in \Sigma_{Full}$ such that $s \overset{a}{\Rightarrow} s^*$ and $(\sigma_1, \sigma_2) \leq^1 (\sigma_*, \sigma_2)$*

Finding 11. *Deception* *- Player 1 (p) deceives Player 2 (q) if for a hypergame H, the following is true:*

- *Player 2 trusts Player 1, according to Finding 5 or Finding 6.*

- *Player 2 misperceives the strategy of Player 1, according to Finding 10.*

- *If there exists a strategy pair (σ_p, σ_q), $\sigma_p \in \Sigma_p$ and $\sigma_q \in \Sigma_q$ where (σ_p, σ_q) $\in$ N(G_T) and (σ_p, σ_q) $\notin$ SHN(G_T) and (σ_{p^*}, σ_q) $\in$ SHN(G_T)*

Finding 12. *In every finite temporal hypergame with mixed strategies, there is at least one SPNE (which may be in the base game).*

Finding 13. *If a temporal hypergame H at some time x has a Nash equilibrium in the base game (i.e. stable hyper Nash equilibrium), then the temporal hypergame has a SPNE.*

Finding 14. *In both the infinite and finite temporal hypergame with mixed strategies, there is at least one SPNE.*

Finding 15. *In a hypergame H, a strategy is a SPNE in the base game iff there is no profitable one-shot deviation. A one-shot deviation would produce a strategy in the difference game of hypergame H.*

8.2 Future Work

The first area would be gathering empirical evidence on how to integrate this approach into real-world decision making problems for the warfighter. The warfighter (especially in the cyber sense) would benefit from the insights and concise image of the battlefield that hypergame theory present such as the temporal hypergame framework presented in this dissertation. This approach has a lot of theoretical applicability, but it is still a model that has not been exercised in any real capacity. Especially when considering the functional payoffs. It is also necessary to understand the ability of the warfighter when using tools such as those proposed in this research. First, is the level of information required for modeling realistic and obtainable? Second, is the warfighter able to use a tool that models misperceptions without introducing additional bias into the decision making process?

The second area of future work concerns the jHALF software for analyzing hypergames [135]. It should be updated to support the temporal hypergame framework proposed in this research. There are indications in the design of the jHALF software or the temporal hypergame framework that would make the two incompatible. Incorporation also opens the door for additional analysis, such as modeling checking. While modeling checking was not part of this research, its applicability to Hypergame Theory should be investigated and understood.

The third area of future research is into real-time strategy games. The hypergame model can be used to determine the best strategy given the game environment. It also be used to train a decision maker by adding or removing columns and rows from the hypergame. This allows game ply by the AI to be tuned to the player's ability - becoming harder for advanced players or softening for novice players. By applying the temporal hypergame framework to the real-time strategy games, complex strategies can be formed to mimic real world events.

The fourth area is proving how "strategy switching" [275], as defined in the temporal hypergame framework, affects the outcomes and analysis. Strategy switching can cause a cycle to appear over time as the game progresses. The properties of the hypergame model should be identified to limit the cycling. This may involve using a subset of the strategy switching operators in the language model.

Appendix A: Hypergame Military Applications Expanded

This appendix expands on the military applications of hypergames. Previously, a short review was provided on many of the applications. Here each application is expanded to show the notion as well as how the analysis was performed by the original authors. The expanded form is contained in an appendix in order to keep flow and consistency in the main body of the document. The Fall of France, Nationalization of the Suez Canal, an Arms Race, Nuclear Confrontation, and the Falkland/Malvinas Conflict are discussed.

A.1 Fall of France

Bennett and Dando [40] [39] first applied hypergames to the first real world application during their analysis of the Fall of France during WWII. Hypergame representation of the Fall of France by Bennett and Dandos is shown in Table A.1.

The Germans believe the French would not include the Germans attacking through the Ardennes forest since such an attack could be stopped. The Germans therefore reasoned the French would believe an attack in the north (Belgian plains) or the south (Maginot Line) was more likely. The French player is only able to see the game on the left in Table A.1. Given the outcomes, the French follows the Nash Equilibrium, choosing F_2 as their strategy and expecting the Germans to choose G_2, but the Germans were playing a metagame that incorporated the broader view, and included the strategy the French had discounted. The Germans figured the French would not consider defending the center heavy, so they decided to attack there, choosing G_3 from the game on the right in Table A.1. This lead to the success of the German attack and France falling quickly. This attack had to be excluded from French rationale in order to make it feasible and allow the Germans to select their highest expected utility based on probable French thought.

206

Table A.1: Hypergame Representations of the Fall of France.

		French	
		F_1	F_2
Germans	G_1	1,4	2,3
	G_2	4,1	3,2*

French				
F_1	F_2	F_3		
1,4	2,3	2,3	G_1	Germans
4,1	3,2	3,2*	G_2	* Nash Equilibrium
3,2	5,0**	2,3	G_3	** True Outcome

A.2 Nationalization of the Suez Canal

Wright et. al. [369] [328] presented a more complex hypergame example in their analysis of the Nationalization of the Suez Canal. By 1955 Egypt was becoming nationalistic, and pursued plans to free itself of control by Britain. Britain wanted to protect Western marine traffic, including safeguarding oil shipped from the Middle East, by preventing Russian interference in the Middle East and protecting the Suez Canal. Tensions had also increased between Egypt and Israel. Military raids between the two countries lead to both countries desiring a stronger military. In August of 1955, Egypt wanted to purchase arms to support its military. France, Britain, and the U.S. did not supply arms because of the Tripartite Declaration of 1950 limited arms deals. This lead to Egypt approaching Russia to purchase arms.

In 1956, Egypt wanted to create hydroelectric power and new farmland by building the High Aswan Dam. The Egyptian government could not afford the cost of constructing the dam without help from either the West or Russia. The West was concerned that Russia would seize the opportunity to fund the construction to develop closer ties with Egypt, building on the momentum of the previous arms deal. Western funding would keep Russia from gaining influence in the Middle East, while allowing Britain to gain influence after its failure to support the arms deal. Meanwhile the U.S. desired to help Egypt develop independence economically.

By December 1955 a proposal to finance the dam construction was made by Britain, the U.S., and the World Bank. The deal provided 30% of the cost of constructing and imposed numerous conditions:

- Egypt had to commit one third of its internal revenue for 10 years to the construction of the dam

- Egypt was required to use economic policy to limit inflation from the addition of foreign capital

- Competitive selection was required for contracts

- Egypt could not accept help from communist countries

Egypt already wanting less control by Britain, feared the terms of the loan would lead to Western dominance, and decided to reject the finance proposal. This caused Egypt to change the proposal and send a counter offer to the West in February 1956. By this time domestic policy in the U.S. was shifting and anti-western sentiment was increasing in the Middle East and the U.S. was quickly losing interest in the financing offer. Meanwhile the General of the Jordan Army, an Englishman, was dismissed which was seen as a political move caused by the anti-western sentiment. This caused the U.S. and Britain to let the finance proposal expire. Egypt hoped either the counter offer would be accepted or its negotiating position would change.

By April 1956, Egypt desired to purchase more weapons as the Israeli attacks increased. A deal was reached in May of 1956 to limit the shipment of arms into the Middle East by Russia, the U.S., and Britain, which lead to Egypt turning to the People's Republic of China. This increased tension with the U.s. and Britain, putting more strain on negotiating the loan offer. Egypt was becoming aware of lack of interest in the loan by the West and decided to accept the original loan offer. In

an effort to get the U.S. to reconsider the original loan, Egypt stated it could also get a loan from Russia. The U.S. felt Egypt was trying to blackmail and withdrew the loan offer with the British, with the official reason being Egypt did not have the resources to complete the dam construction.

Egypt turned to Russia for a last chance at a loan, but Russia offered no such offer. Without foreign capital, seizure of the Suez Canal was attractive to Egypt. Nationalization of the Canal would raise much needed cash and remove the last of Western control in Egypt.

Egypt wanted to avoid another ultimatum from the West and decided for a surprise canal takeover. A surprise takeover was thought to help avoid the loss of life from military clashes. Egypt also thought it would take Britain two months to prepare a military response and a settlement could be negotiated before the response took place. In July of 1956, Egypt nationalized the Suez Canal, denouncing the West in the process. the West was shocked by the move, and immediately raised concerns about the security of the canal.

The hypergame is formalized by picking a point in time to model. The authors choose February of 1956 when Egypt proposed an alternative loan agreement which is before the West became discouraged with the process. Egypt is trying to assert its nationalism and wants to finance the Aswan Dam. Britain wants to be influential in the Middle East, preventing Russian influence and appealing Egypt after the arms refusal. The U.S. wanted to limit Russia's influence in the Middle East and promote Egyptian nationalism without upsetting Britain. Russia wanted stronger influence in the Middle East.

Britain and the U.S. are modeled as one player, since they act together on the loan proposal. Because Russia is not an active participant, it is not represented in the game. The options of the hypergame are the following:

Britain and U.S.

1. Offer loan based on original conditions

2. Offer loan on Egypt's conditions

Egypt

3. Negotiate loan based on original conditions

4. Negotiate loan on Egypt's conditions

5. Appease West

6. Pursue Russian loan.

Egypt (secret)

7. Pursue Russian loan and if it fails, then nationalize the Suez Canal

First the unfeasible options are excluded such as the West offering a loan on the original conditions and offering a loan with Egypt's conditions. The possible outcomes are listed in Table A.2 where each column is an outcome. Each outcome is decimalized for easy manipulation. Decimalization is accomplished by treating each outcome as a binary number with the lowest order bit on top. The binary number is then converted to a decimal number.

Decimalized outcomes are then put in order of preference for each player. The most preferred outcome is placed on the left, with the least preferred on the right, as shown in Table A.3. Two preference vectors are used for Egypt - one to show Egypt's preferences from the viewpoint of the West and the other for Egypt's real preferences. The only difference between vectors is that the first vector does not contain option 7.

Each outcome is then analyzed for stability from the point of view of each player. The equilibria are formed from outcomes that are stable for all of the players. This is

Table A.2: Possible Outcomes.

Players/Options							Possible Outcomes									
							West									
1)	0	1	0	1	0	1	0	0	1	0	1	0	0	1	0	1
2)	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0
							Egypt									
3)	0	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0
4)	0	0	0	0	1	1	1	0	0	1	1	1	0	0	0	0
5)	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
6)	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
							Egypt(secret)									
7)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Decimalized	0	1	4	5	8	9	10	20	21	24	25	26	32	33	64*	65*

* Unknown to the West

done by looking for Unilateral Improvements (UIs) from each outcome in a player's preference vector. Unilateral Improvement (UI) is formed when the player reaches a preferred outcome where the other player's strategies remain the same. Table A.4 give the UIs for each outcome in the preference vector. If an outcome does not have a UI then it is rational and stable for the given player. In the table "r" indicates rational outcomes, "u" indicates an unstable outcome. Unstable outcomes exist when there is a UI where another player cannot make an improvement which results in an outcome worse than the original outcome for the player. All UIs below an outcome must be checked to see if another player can deter the outcome. The outcome is stable, marked by an "s", if all of the UIs have a deterrent. For example, Egypt has a UI from 65 to 5. The West does not realize 65 is a possibility, but there is an UI from 5 to 4. Since

4 is less preferred to 65 by Egypt, this deters Egypt from moving from outcome 65 to 5.

Equlibria are where outcomes are stable for all players. Egypt's and the West's preference vectors are compared, if both vectors have an "r" or "s" then an "E" is placed above the preference to denote an equilibrium. An "x" denotes a lack of equilibrium. Outcomes 5, 64, and 65 are the true equilibria to the Suez Crisis, while the West believes 5, 32, and 33 are the equilibria. This shows outcome 5 is preferred by all players - Egypt would accept the original loan. From history it is known, this is not what happened. Egypt was not irrational. Instead Egypt tried to wait for the West's attitude to change and allow outcome 10 - a loan on Egypt's terms. The West's attitude did not change in Egypt's favor, and the possibility of a loan passed. This left only 64 and 65 as equilibria, and the outcome 64 is what happened in history.

A.3 Arms Race

Bennett and Dando [41] also model an arms race as a hypergame where they model an arms race between two nations, Nation A and Nation B. Higher numbers are used to represent more highly preferred outcomes, as shown in Figure A.1. The preference of both nations is peace loving with the following preferences: (4) mutual disarmament, (3) arms lead for self, (2) arms Race, and (1) arms lead for opponent. In this game, both peace loving nations would have no trouble reaching the desired outcome of mutual disarmament. Mutual disarmament is the only stable outcome (both nations choose "disarmâĂŹâĂŹ).

Even with peace loving players, Bennett and Dando introduce misperception into the model by giving NationâĂŹs X belief of the opponent preferences. Each nation believes its opponents preferences are the following: (4) arms lead for self, (3) mutual disarmament, (2) arms race, and (1) arms lead for opponent. As shown in

Figure A.1: Arms Race Model âĂŞ Preferred Outcome.

Figure A.2, these misperceptions about opponent preferences leads to a hypergame where each nation perceives a slightly different version of the same game. While each nation would like to move to mutual disarmament, they will be deterred because of the perception that the other nation would prefer to an arms lead. This leads to an arms race (both nations choose "armâĂŹâĂŹ).

The authors give two reasons for the nations to have misperceptions: (1) failure to see how the other side sees the world, i.e. alternative views, (2) concentration on the individual instead of the system in general. Their analysis forces the analyst to consider the perceptions, beliefs, and actions of all parties involved, which they claim leads to a more competent analysis.

A.4 Nuclear Confrotation

Fraser et al. [107] apply five conflict analysis models to a possible nuclear confrontation between the USA and USSR. The five conflict analysis models are

213

Figure A.2: Arms Race Model âĂŞ Misperceptions introduce a hypergame.

normal form analysis from game theory, metagame analysis [175], and hypergame analysis [108] [106]. An overview of each of the models follows.

The normal form model is shown in Figure A.3 and is taken from the work of Richelson [304]. The game theoretic model is played between two players, United States of America (USA) and Union of Soviet Socialist Republics (USSR). The three strategies for each country are: (C) conventional attack, (L) limited nuclear strike, (S) full nuclear attack.

The authors state that the normal form is useful for giving structure to real world problems, as well as modelling the interactions between players. They also concluded the normal form lacks ability to model complicated problems, including ones with more than two players or a large number of strategies, and is not convenient for solving the model for equilibria that are more subtle [107]. In order to overcome the issues with normal form, the authors use metagame theory from Howard [175].

Figure A.3: USA âĂŞ USSR Nuclear Confrontation in Normal Form.

Metagames are game theoretic models that take into account the possible reactions a player will have to the known strategies of another player(s). The playerâĂŹs strategies reflect its reaction to the other player(s) in a metagame, instead of just the actions the player can choose from in the normal form. Because players can have reaction to reactions, there are an infinite number of metagames for each basic game. To reduce the number of metagames, and allow for analysis, Howard ([175] developed the Characterization Theorem. The Characterization Theorem allows analysis of all possible metagames, while only analyzing the initial game. The outcomes of the initial metagame are known as metarational outcomes, and either is considered stable for a particular player. There are three types of metarational outcomes:

- Rational Outcomes (Nash Equilibrium)

- Symmetric Metarational Outcomes (Dominated Strategies)

215

- General Metarational Outcomes (Weak form of stability)

Using the basic game shown in Figure A.3, the metagame analysis shown in Figure A.4 is constructed for the USSR outcome (L, C) or (010 100) or limited nuclear attack by USA and conventional attack by USSR. The player and the actions available to each are listed on the left side of the figure. Next a 1 or 0 is placed in the same row as the actions to indicate if the action is taken (1) or rejected (0). The combination of 1âĂŹs and 0âĂŹs in a column forms a strategy for the particular player. Each feasible outcome is placed either in the preferred category or the not preferred category relative to the particular outcome under analysis.

USA	Preferred					Particular Outcome	Not Preferred		
C	1	1 1 0 0				0	0	0 0
L	0	0 0 1 1				1	0	0 0
S	0	0 0 0 0				0	1	1 1
USSR								
C	0	1 0 0 0				1	0	0 1
L	1	0 0 1 0				0	0	1 0
S	0	0 1 0 1				0	1	0 0

unilateral improvements

Inescapable sanctions

Figure A.4: Metagame Analysis of Outcome (010 100) for the USSR.

This outcome is symmetric metarational for USSR because USA has the ability to enforce sanctions on the USSR. The stability for the all the outcomes in this game are listed in Table A.5. This shows that the outcome of total nuclear war is rational. The other four possible outcomes depend on whether the USA can enforce the sanctions, due to the symmetric metarationality for the individual strategies. The ability to

216

determine if sanctions can be enforce is a complex problem, for which the authors do not give a solution. This leads to the majority of outcomes being equilibria.

The improved metagame analysis proposed by Fraser and Hipel [108] [106] allow for hypergame analysis. The improved analysis uses the known preferences of the other player(s) to test if the threat of sanction is credible. The authors convert the binary preference vector into a decimalized form. For example, the outcome (100 010) has a decimalized value of: $1 * 2^0 + 0 * 2^1 + 0 * 2^2 + 0 * 2^3 + 1 * 2^4 + 0 * 2^5 = 1 + 16 = 17$. The three distinct representations and the relationship for outcomes are shown in Table A.6.

Next the UIs of every player for every one of the outcomes in the preference vectors are found. UIs are recorded below the preference vector outcomes in decreasing order of preference. This is shown in Table A.7. Above each outcome is recorded the player stability: r for rational, s for symmetric metarational, and u for unstable. Then overall stability is identified by E for equilibrium and X for unstable. Again, the analysis shows the only stable outcome that is rational for both players is 36, or (001 001), or (S,S).

Their analysis determine the improved metagame analysis (i.e. hypergame analysis) of conflicts is the best for modeling real world conflicts. This is due to its nature for extension, where it can be used to model coalitions and bargaining/negotiation situations between players.

A.5 Falkland/Malvinas Conflict

Hipel et al. [167] examine the Falkland/Malvinas conflict between Britain and Argentina in 1982. The authors approach the conflict from a different angle in their analysis of the conflict between Britain and Argentina. The hypergame analysis of the conflict is used to show how misperceptions dictated an outcome that was unexpected

by all sides. The authors construct the hypergame model based on historical material, using a first-level hypergame.

The conflict can only be modeled at a certain point and time because the conflict is a dynamic phenomenon. A player's perceptions, actions, strategies, and preferences will change over the course of the conflict. The authors therefor chose three important periods of time in the conflict: the day of invasion by Argentina, a month long period of negotiations, and the day Britain issued a military response. Each period of time is modeled as a game iteration where each choice and outcome from a previous iteration affects the next iteration.

The game changes at each iteration as the choices of the players are used to updated the hypergame model, as shown in Table A.8. In the table, an underlined strategy refers to the strategy taken by the corresponding player. As the hypergame model is updated, it is possible to reduce it to a simple game. The simple game forms as misperceptions are corrected as players become aware of the remaining actions and outcomes. It is shown that the stability analysis for Britain is the belief that both sides would prefer to settle the conflict by peace. Argentina's preference is to invade. This gives the stability set {settle by peace, invade}. The real outcome is not expected by either player.

At the next iteration Argentina continues to play the same game while Britain updates its game with Argentina's past actions. Britain tried to maximize the pressure for a peaceful resolution, while Argentina continued to view the British warning as a bluff. At this point if both player's have the same attitude (and information) the conflict could be avoided.

At the third iteration it can be seen that the player's knowledge has increased since the start of the conflict, although misperceptions by both players played a fatal role at each iteration. At this point the player's had complete information about

218

the game and the other player's preferences, but the conflict had already moved to a point where neither player could return to their original position. Therefore, the worst possible outcome, war, happens in the conflict.

Table A.3: Preference Vectors.

Egypt	10	5	[64	65]	[32	33]	[0	1	4	8	9]	[20	24	25]	26	21
West's View of Egypt	10	5			[32	33]	[0	1	4	8	9]	[20	24	25]	26	21
West	21	[20	24	25]	[0	1	4	8	9]	5	26	[32	33]	10		

[] = equally preferred outcomes

Table A.4: UIs in Preference Vectors.

Egypt's View of Egypt

X	E	E	E	X	X	X	X	X	X	X	X	X	X	X	X
r	r	r	s	u	u	u	u	u	u	u	u	u	u	u	u
10	5	[64	65]	[32	33]	[0	1	4	8	9]	[20	24	25]	26	21
			5	64	5	64	5	64	64	5	64	64	5	10	5
				65	32	65	32	32		65	32	32	65		65
				33		33					8	8	33		33
											4	4	1		1
													9		9
															25

British and American View of Egypt

X	E	E	E	X	X	X	X	X	X	X	X	X	X
r	r	r	s	u	u	u	u	u	u	u	u	u	u
10	5	[32	33]	[0	1	4	8	9]	[20	24	25]	26	21
			5	32	5	32	32	5	32	32	5	10	5
				33					33	8	8	33	33
										4	4	1	1
												9	9
													25

Britain and U.S.

r	s	r	r	r	r	r	r	r	s	u	r	r	u
21	[20	24	25]	[0	1	4	8	9]	5	26	[32	33]	10
	21								4	24			9
										25			8

221

Table A.5: Metagame Analysis Results for the USA âĂŞ USSR Nuclear Confrontation.

Outcome	USA Stability	USSR Stability	Overall Stability
(100 100)	Symmetric Metarational	Symmetric Metarational	Equilibrium (if credible)
(100 010)	Symmetric Metarational	Rational	Equilibrium (if credible)
(100 001)	Unstable	Symmetric Metarational	Unstable
(010 100)	Rational	Symmetric Metarational	Equilibrium (if credible)
(010 010)	Symmetric Metarational	Rational	Equilibrium (if credible)
(010 001)	Unstable	Symmetric Metarational	Unstable
(001 100)	Symmetric Metarational	Unstable	Unstable
(001 010)	Rational	Unstable	Unstable
(001 001)	Rational	Rational	Equilibrium

Table A.6: Outcome Representations and Relationships [107].

Players/Options	Outcomes								
USA									
C	1	0	0	1	0	0	1	0	0
L	0	1	0	0	1	0	0	1	0
S	0	0	1	0	0	1	0	0	1
USSR									
C	1	1	1	0	0	0	0	0	0
L	0	0	0	1	1	1	0	0	0
S	0	0	0	0	0	0	1	1	1
Normal Form:	(C,C)	(L,C)	(S,C)	(C,L)	(L,L)	(S,L)	(C,S)	(L,S)	(S,S)
Decimalized Outcomes	9	10	12	17	18	20	33	34	36

Table A.7: Improved Metagame Stability Analysis [107].

	E	X	X	E	E	X	E	X	X	overall stability
	r	s	r	s	s	u	r	u	u	player stability
USA	10	12	20	18	9	17	36	34	33	preference vector
		10		20	10	20		36	36	UIs
					12	18			34	
	r	s	s	r	s	s	r	u	u	player stability
USSR	17	9	33	18	34	10	36	20	12	preference vector
		17	17		18	18		36	36	UIs
			9			34			20	

223

Table A.8: Perceptual and overall equilibria in the Falkland Islands crisis.

Game Equilibria	$H\prime$ (Iteration 1)	$H\prime$ (Iteration 2)	$H\prime$ (Iteration 3)
Argentinian	$\begin{bmatrix} \underline{Invade} \\ Blockade \end{bmatrix}$	$\begin{bmatrix} \underline{Maintain} \\ Blockade \end{bmatrix}$	
British	$\begin{bmatrix} SettlebyPeace \\ SettlebyPeace \end{bmatrix}$	$\begin{bmatrix} Withdraw \\ Negotiate \\ \underline{Negotiate} \end{bmatrix}$ $\begin{bmatrix} Withdraw \\ \underline{Blockade} \end{bmatrix}$ $\begin{bmatrix} Maintain \\ \underline{Blockade} \\ \underline{Invade} \end{bmatrix}$	
Overall	$\begin{bmatrix} Invade \\ SettlebyPeace \end{bmatrix}$	$\begin{bmatrix} Maintian \\ Negotiate \end{bmatrix}$ $\begin{bmatrix} Maintain \\ Blockade \end{bmatrix}$ $\begin{bmatrix} Maintain \\ Blockade \\ Invade \end{bmatrix}$	$\begin{bmatrix} Maintain \\ Invade \end{bmatrix}$

224

Appendix B: Hypergame Network Defense Application Expanded

There are many approaches that apply game theory to intrusion problems on networks. One approach uses game theory to select which security measures are applied to the network based on which ones result in the lowest incurred cost while providing the highest level of security, thus allowing the system to be optimized [326]. This allows game theory to be a worthwhile tool in influencing the decisions made in the deployment and use of intrusion detection systems. Research similar to [326] shows the ability of game theory to solve and optimize network intrusion problems. This section focuses on two key research efforts that have expanded the modeling ability of intrusion detection.

B.1 Game Theoretic Model

Chen and Leneutre [72] model a heterogeneous network for intrusion detection using game theory. They use the classic two-player game with an attacker and a defender; each player has two strategies: to attack/not attack or defend/not defend. The model presented by Chen and Leneutre starts with a network, $\mathcal{N} = (S_D, S_A, \mathcal{T})$. S_D is the set of defending agents with an IDS module and S_A is the set of attackers, where $\mathcal{T} = \{1,2,...,N\}$ is the set of targets or network nodes that can be attacked. The normal form of the intrusion model is shown in Table B.1

Table B.1: Chen and Leneutre Intrusion Model Normal Form for Target i.

	Monitor	Not Monitor
Attack	$(1 - 2a)W_i - C_aW_i$, $-(1 - 2a)W_i - C_mW_i$	$W_i - C_aW_i$, $- W_i$
Not Attack	0, $-bC_fWi - C_mW_i$	$0, 0$

The attacker attempts to attack the target nodes without being detected by choosing a strategy, p $= \{p_1, \ p_1, \ ..., \ p_N\}$,according to the attack probability distribution of the target set, $\mathcal{T}$, where p represents the probability of attacking target i, and $\sum_{i \in \mathcal{T}} p_i \leq P \leq 1$ is the resource constraint of the attacker. The defender monitors the target nodes by choosing a strategy, q $= \{q_1, q_1, ..., q_N\}$, where $\sum_{i \in \mathcal{T}} q_i \leq Q \leq 1$ is the defender's resource constraint.

The player's utility values are based on functions with predetermined variables, as shown in Table B.2. W_i represents the loss of security on a node. The detection rate of the defender's IDS is denoted a and the false alarm rate is denoted b, where $a, \ b \in [0,1]$. The cost of attacking is represented by $C_a W_i$, the cost of monitoring represented by $C_m W_i$, and $C_f W_i$ is the loss of a false alarm. The model assumes $C_a < 1$ and $C_m < 1$, so the players have incentive to monitor and attack.

Table B.2: Predetermined Variables from Chen and Lenutre.

Variable Symbol	Meaning
a	IDS Detection Rate
b	IDS False Alarm Rate
C_a	Cost of Attack
C_f	Cost of False Alarm
C_m	Cost of Monitoring
W_i	Value of Target

The payoffs of the attacker is given by U_A and the payoffs of the defender is given by U_D:

$$U_A(p, q) = \sum_{i \in \mathcal{N}} p_i W_i (1 - 2aq_i - C_a)$$

$$U_D(p, q) = \sum_{i \in \mathcal{N}} q_i W_i [p_i(2a + bC_f) - (bC_f + C_m)] \text{ - } \sum_{i \in \mathcal{N}} p_i W_i$$

This results in a intrusion detection game with an attacker and defender defined with the following properties:

- Players: Attacker, Defender

- Strategy sets:

  - Attacker: $A_A = \{\text{p:p} \in [0, P]^N, \sum_{i \in \mathcal{N}} p_i \leq P\}$
  - Defender: $A_D = \{\text{q:q} \in [0, Q]^N, \sum_{i \in \mathcal{N}} q_i \leq Q\}$

- Payoff: U_A for attacker and U_D for defender

- Game Rule: Strategy selection is done by the attacker/defender according to p/q $\in A_A/A_D$, maximizing U_A/U_D

The authors use the utility values based on variables to show a real world intrusion problem, where the rational attacker would have preference to attack higher value targets. This is shown through Nash equilibrium analysis, as the mixed strategy Nash equilibrium is zero with the target is not in the rational set. This research contributes to intrusion detection by showing that increasing attack or monitoring does not affect the Nash equilibrium, or does the attacker gain from a decreased attack cost. Cost decreases are demonstrated when the intrusion detection system improves its performance through improving the detection rate of intrusions or decreasing the rate of false alarms. The authors use of variables in utility functions allow for valuable insight into how to improve the individual player's performance.

B.2 Hypergame Model

Alan Gibson presents a model based on the intrusion model presented by Chen and Leneutre [72] and the Hypergame Normal Form model presented by Vane [356]

[355]. The author achieves a model that has a changeable nonzero-sum utility values with a process for delineation of strategy selection [135]. In order to achieve this model, the Chen and Leneutre intrusion model is extended by adding strategies for both the attacker and defender; while the HNF model is used to hide or discount strategies from the other player.

In order to use hypergame theory, the game is organized with the defender as row, as the Hypergame Normal Form analyzes the game from the perspective of the row player. The model also keeps the functional and nonzero-sum utilities from the Chen and Leneutre model. The resulting game model is shown in Figure B.1.

Attacker Defender	Not Attack		Attack		Zero-Day Exploit	
Not Defend	0	0	$-W_i$	$W_i - C_a W_i$	$-W_i$	$W_i - C_z W_i$
Defend	$-bC_f W_i - C_m W_i$	0	$-(1-2a)W_i - C_m W_i$	$(1-2a)W_i - C_a W_i$	$-W_i$	$W_i - C_z W_i$
Provide Ruse	$-W_i - C_r W_i$	0	$V_a - C_r W_i$	$W_i - C_a W_i$	$-W_i$	$W_i - C_z W_i$
Shut Down	$-W_i - C_t W_i$	0	$V_a - W_i - C_t W_i$	$-W_i$	$V_a - C_t W_i$	$-W_i$

Figure B.1: Game Model as Presented by [135].

In order to use HNF as presented by Vane, the game is organized with the defender as row, as the Hypergame Normal Form analyzes the game from the perspective of the row player. The model also keeps the functional and non-zero-sum utilities from the Chen and Leneutre model. The resulting game model is shown in Figure B.3. The attacker is given a new strategy, zero-day exploit, which is an attack where there is no defense since the vulnerability is undiscovered. The defender is given two new strategies: provide ruse or shutdown. A defender may provide a ruse by fooling the attacker into attacking a honeypot, while collecting information about

the type and style of the attack. The shutdown option allows the defender to remove the system from the network and stop the attack in its tracks, but also removes the system from operation even for mission critical activities.

Table B.3: Gibson Intrusion Model Normal Form for Target i.

	Not Attack	Attack	Zero-Day Exploit
Not Defend	0, 0	$-W_i$, $W_i - C_a W_i$	$-W_i$, $W_i - C_z W_i$
Defend	$-bC_f W_i - C_m W_i$, 0	$-(1\text{-}2a)W_i - C_m W_i$, $(1\text{-}2a)W_i - C_a W_i$	$-W_i$, $W_i - C_z W_i$
Provide Ruse	$-W_i - C_r W_i$, 0	$V_a - C_r W_i$, $W_i - C_a W_i$	$-W_i$, $W_i - C_z W_i$
Shut Down	$-W_i - C_t W_i$, 0	$V_a - W_i - C_t W_i$, $-W_i$	$V_a - C_t W_i$, $-W_i$

New variables are added for the calculation of utility payoffs, as shown in Table B.4 with the original Chen and Leneutre variables. A zero-day exploit has a cost of C_z which is a percentage of the value of the target. The shutdown option has a cost of C_t which is the deration of time the system is unavailable on the network. The provide ruse option has a cost of C_r which is the time or sophistication level of the ruse. The important variable introduced by Gibson is the value of the attacker, V_a which allows certain strategies to be more worthwhile as the complexity and danger level of the attacker increases. This allows the game to represent different levels of attackers, such as script kiddies, hackavists, and national states.

Since the variables are not static and change over time or iterations an initial value and a method for updated the variables is needed. The initial values for the variables introduced by Gibson are shown in Table B.5. The variables are updated using a predetermined change amount. The amounts for the variables introduced by Gibson are shown in Table B.6.

229

Table B.4: Enhanced Model Variables from Gibson.

Variable Symbol	Meaning
a	IDS Detection Rate
b	IDS False Alarm Rate
C_a	Cost of Attack
C_f	Cost of False Alarm
C_m	Cost of Monitoring
C_r	Cost of Providing Ruse
C_t	Cost of Time Down
C_z	Cost of Zero-Day Exploit
V_a	Value of Attacker
W_i	Value of Target

The payoffs of the attacker is given by U_A and the payoffs of the defender is given by U_D:

$$U_A(p,q) = \sum_{i \in \mathcal{N}} \left(p_i^A(1-q_i)[W_i - C_aW_i] + p_i^Z(1-q_i)[W_i - C_zW_i] + p_i^A q_i^D[(1\text{-}2\text{a})W_i - C_aW_i] + p_i^Z q_i^D[W_i - C_zW_i] + p_i^A q_i^P[W_i - C_aW_i] + p_i^Z q_i^P[W_i - C_zW_i] + p_i^A q_i^S[-W_i] + p_i^Z q_i^S[-W_i] \right)$$

$$U_D(p,q) = \sum_{i \in \mathcal{N}} \left(p_i^A(1-q_i)[-W_i] + p_i^Z(1-q_i)[-W_i] + (1\text{+}p_i)q_i^D[\text{-b}C_fW_i - C_mW_i] + p_i^A q_i^D[-(1\text{-}2\text{a})W_i - C_mW_i] + p_i^Z q_i^D[-W_i] + (1\text{+}p_i)q_i^P[-W_i - C_rW_i] + p_i^A q_i^P[V_a - C_rW_i] + p_i^Z q_i^P[\text{-} W_i] + (1 - P_i)q_i^S[-W_i - C_tW_i] + p_i^A q_i^S[V_a - W_i - C_tW_i] + p_i^Z q_i^S[V_a - C_tW_i] \right)$$

This results in a intrusion detection game with an attacker and defender defined with the following properties:

Table B.5: Gibson's defender type initial variable values.

Variable	Nuisance	Low	Mid	High	All-Out
Detection rate for attack (a)	0.90	0.88	0.85	0.83	0.80
False alarm rate (b)	0.02	0.03	0.04	0.05	0.06
Cost of attack (C_a)	0.10	0.15	0.20	0.25	0.30
Cost of false alarm (C_f)	0.90	0.85	0.80	0.75	0.70
Cost of monitoring (C_m)	0.10	0.15	0.20	0.25	0.30
Cost of providing ruse (C_r)	1.00	0.75	0.50	0.25	0.00
Cost of time down (C_t)	2.00	1.50	1.00	0.50	0.00
Cost of zero-day exploit (C_z)	1.00	1.00	1.00	0.80	0.50
Value of Attacker (C_z)	0.00	1.00	2.00	3.00	4.00
Value of Target (C_t)	1.00	1.00	1.00	1.00	1.00

- Players: Attacker, Defender

- Strategy sets:

    - Attacker: $A_A = \{p{:}p \in [0, P]^N, \sum_{i \in \mathcal{N}} p_i \leq P\}$
    - Defender: $A_D = \{q{:}q \in [0, Q]^N, \sum_{i \in \mathcal{N}} q_i \leq Q\}$

- Payoff: U_A for attacker and U_D for defender

- Game Rule: Strategy selection is done by the attacker/defender according to p/q $\in A_A/A_D$, maximizing U_A/U_D

B.3   Conclusion

A unique part of the model is that the attacker's utility is the same for the strategy to attack when the defender selects either the not defend or provide ruse strategies. This correctly models the deployment of a sound honeypot where the

Table B.6: Gibson variable update changes.

Variable ID	Variable Name	Strategy Selected Defender	Strategy Selected Attacker	Value Changes Amount
C_r	Cost of	Not Defend	Any	-0.5
	Providing Ruse	Defend	Any	-0.5
		Provide Ruse	Any	0.1
		Shut Down	Any	0.1
C_t	Cost of	Not Defend	Any	-0.02
	Time Down	Defend	Any	-0.02
		Provide Ruse	Any	0.5
		Shut Down	Any	1.0
C_z	Cost of	Not Defend	Zero-Day	0.1
	Zero-Day	Defend	Zero-Day	0.4
		Provide Ruse	Zero-Day	0.1
V_a	Value of	Not Defend	Not Attack	-0.01
	Attacker	Defend	Not Attack	-0.01
		Provide Ruse	Not Attack	-0.01
		Not Defend	Attack	0.5
		Defend	Attack	0.1
		Provide Ruse	Attack	0.5
		Not Defend	Zero-Day	0.5
		Defend	Zero-Day	1.0
		Provide Ruse	Zero-Day	1.0

attacker believes they have succeeded in attacking the desired system. It also gives the attacker decreased utility when the system is shut down because of an attack. This incorporates a rational attacker's preference to keep system up and running in order to continue collecting information and prolonging the attack. The shutdown system strategy is modeled as the only effective strategy against the zero-day exploit strategy. The zero-day exploit strategy is considered costly to an attacker since they are generally costly to find and once used are generally fixed so they no longer work. Given the attacker does not consider the new defender strategies and the attacker's zero-day exploit is too costly, the subgame that results is the original Chen and Leneutre model.

The most important contribution of Gibson's model is that by combining the Chen and Leneutre model with HNF, dynamic variables are added to the payoff functions in HNF. This allows for dynamic play and updating of variables as the game is played.

Appendix C: Prisoner's Dilemma Temporal Hypergame Example

This appendix demonstrates the application of the temporal hypergame framework to the classical game theoretic example of the Prisoner's Dilemma. The framework is applied to the Prisoner's Dilemma in order to show the framework is valid with classical games. The structural properties of the game do not allow for detailed hypergame analysis, since the game is symmetric and does not have differences in player perceptions required for hypergame analysis.

In game theory, the Prisoner's Dilemma is a classic example of how the interaction between two individuals leads to cooperation or not. In the Prisoner's Dilemma there are two players, Prisoner A and Prisoner B [90, 104, 284]. Each player can choose one of two actions, either the player can choose to cooperate by staying silent or defect by betraying the other player.

The consequence of selecting an action results in no jail time (jail(0)) or jail time (jail(x) where x is the amount of time in jail). The payoff function is if the player cooperates and the other player cooperates, both receive one year in jail, otherwise if the other player defects the player receives 10 years in jail while the other player receives no jail time. If both players defect, then both players receive five years in jail. This game is shown in normal form in Figure C.1. Note that cooperate is abbreviated "C" and defect as "D".

In a single observation Prisoner's Dilemma it is assumed the prisoners do not have an opportunity to punish or reward their partner over the outcome of the game. The only punishment are the prison sentences each prisoner receives and reputations remain intact. After this non-repeated game each prisoner do not have any other interaction with each other.

Figure C.1: Prisoner's Dilemma.

There is a repeated, or iterative, version of the Prisoner's Dilemma where the same prisoners play the game over and over [115]. Since the same players are playing the same multiple times in succession, they can remember their opponent's previous actions and change their strategy accordingly. In this case it is assumed the players know the number of times the game will be played, so the game is finite.

C.1   Initial Game Definition

The repeated Prisoner's Dilemma can be represented using the notation of of the temporal hypergame framework in Section 6.3 of Chapter 6, where the hypergame is defined as:

$$H_{PD}^T \triangleq \{G_{P1}^T\} \tag{C.1}$$

The perceived game G_{PD}^T:

$$G_{PD}^T \triangleq (N, \Sigma_{Full}, \Phi_{PD}^T, \{\preceq^i\}_{i \in N}) \tag{C.2}$$

where N is the set of players, such that N = {Prisoner A, Prisoner B}. Σ is the set of actions available to the prisoners, where $\Sigma = \Sigma_{PrisonerA} \cup \Sigma_{PrisonerB}$. The actions available to Prisoner A is represented by $\Sigma_{PrisonerA} = \{C, D\}$ and Prisoner B by $\Sigma_{PrisonerB} = \{C, D\}$.

C.2 Identification of States and Transitions

The next step to represent a game in the temporal hypergame framework is to identify the states and transitions for the game. The game arena is given by Equation 27 in Chapter 6:

$$\Phi_{PD} = (W, \longrightarrow, w_0, \chi) \tag{C.3}$$

W consists of seven game states, including start, A_C, A_D, $A_C B_C$, $A_C B_D$, $A_D B_C$, and $A_D B_D$. The $\longrightarrow$ defines the game transitions such that $(W \times \Sigma) \rightarrow W$. The possible transitions are:

- start x C $\rightarrow A_C$ where Cooperate $\in \Sigma_{PrisonerA}$

- start x D $\rightarrow A_D$ where Defect $\in \Sigma_{PrisonerA}$

- A_C x C $\rightarrow A_C B_C$ where C $\in \Sigma_{PrisonerB}$

- A_D x C $\rightarrow A_D B_C$ where C $\in \Sigma_{PrisonerB}$

- A_C x D $\rightarrow A_C B_D$ where D $\in \Sigma_{PrisonerB}$

- A_D x D $\rightarrow A_D B_D$ where D $\in \Sigma_{PrisonerB}$

- $A_C B_C \rightarrow$ start

- $A_D B_C \rightarrow$ start

- $A_C B_D \rightarrow$ start

- $A_D B_D \rightarrow$ start

The initial state of the game w_0, is equal to start $\in W$, where $\overrightarrow{start} = \{A_C, A_D\}$. χ - W $\longrightarrow$ N function assigns he player whose turn it is to the game state w $\in$ W. For example, $\chi(start) =$ Prisoner A, while $\chi(A_C) = \chi(A_D) =$ Prisoner B. The preference ordering function for the i_{th} player $\preceq^i = u_i(x, y) \leq u_i(x', y)$ for i $\in$ N. This means for Prisoner A $\preceq^{PrisonerA}$ indicates the outcome (x,y) is preferred to outcome (x', y) if x $\leq x'$ $\forall (x, y) \in W$. For Prisoner B $\preceq^{PrisonerB}$ indicates the outcome (x,y) is preferred to outcome (x, y') if y $\leq y'$ $\forall (x, y) \in W$. Notice the Nash equilibrium concept is encoded in the preference relation, $\preceq^i$, assuming both players are rational.

C.3 Game Mapping

The next step of defining a game in the temporal hypergame framework is to map the game to the extensive form game tree. An extensive form game tree $\mathcal{T}$ can be associated with the Prisoner's Dilemma game arena, ϕ_{PD}. The extensive form game tree is defined from Equation 29 in Chapter 6:

$$\mathcal{T} = (S, \Rightarrow, s_0, \lambda) \tag{C.4}$$

where (S, $\Rightarrow$) is a countably infinite tree rooted at s_0 with edges from Σ. The nodes of the tree are given by S, where S = $\{$root, A_C, A_D, $A_C B_C$, $A_C B_D$, $A_D B_C, A_D B_D\}$. The root of the tree denoted s_0, is equal root $\in$ S. The $\overset{x}{\Rightarrow}$ is the function that moves between nodes of the tree using the edge denoted by x $\in \Sigma$. The possible nodes transitions are root $\overset{C}{\Rightarrow} A_C$, root $\overset{D}{\Rightarrow} A_D$ where C, D $\in \sigma_{PrisonerA}\} \cup$ $\{A_C \overset{C}{\Rightarrow} A_C B_C$, $A_C \overset{D}{\Rightarrow} A_C B_D$, $A_D \overset{C}{\Rightarrow} A_D B_C$, $A_D \overset{D}{\Rightarrow} A_D B_D$ where C, D $\in \sigma_{PrisonerB}$

- λ - S $\rightarrow$ W, where

    - $\lambda(s_0) = w_0$

- $\forall s, s' \in S$, if s $\overset{a}{\Rightarrow} s'$ then $\lambda(s) \overset{a}{\Rightarrow} \lambda(s')$

- if $\lambda(s) = w$ and $w \overset{a}{\rightarrow} w'$ there exists $s' \in S$ s.t. s $\overset{a}{\Rightarrow} s'$ and $\lambda(s') = w'$

C.4 Path Structuring

The model for the Prisoner's Dilemma can be represented by $M_{PD} = (\mathcal{T}, \mathcal{V})$ using the notation discussed in Sections 6.3.1 and 6.3.2. The game tree, $\mathcal{T}$ is given above and the valuation function $\mathcal{V}$ is given by:

- $\mathcal{V}(p_{int}) = \{s_0\}$

- $\mathcal{V}(p_{dom}) = \{A_D,\ A_D B_D\}$

- $\mathcal{V}(p_{worst_A}) = \{A_C,\ A_C B_D\}$

- $\mathcal{V}(p_{worst_B}) = \{A_D,\ A_D B_C\}$

C.5 Define Player Strategies

The next step is to define the player strategies using the temporal hypergame framework constructs defined in Sections 6.3.3 and 6.3.4 of Chapter 6. In this game Prisoner's A strategy can be defined as:

$$Strat^A \equiv ([p_{int} \mapsto D]^A) \cdot ([p_{int} \mapsto C]^A) \cdot ([\boxdot\ [turn_B \mapsto C]^B \Rightarrow [turn_A \mapsto C]^A) \cdot ([\lozenge$$
$$[turn_B \mapsto D]^B \Rightarrow [turn_A \mapsto D]^A)$$

Prisoner's B strategy can be defined as:

$$Strat^B \equiv ([p_{int} \mapsto D]^A \Rightarrow [p_{dom} \mapsto D]^B) \cdot ([p_{int} \mapsto C]^A \Rightarrow [p_{worst_A} \mapsto D]^B) \cdot$$
$$([\boxdot[turn_A \mapsto C]^A \Rightarrow [turn_B \mapsto C]^B) \cdot ([\lozenge[turn_A \mapsto D]^A \Rightarrow [turn_B \mapsto D]^B)$$

C.6 Analyze Model

Thus, $Strat^A \leadsto_B (p_{dom} \vee p_{worst_A})$. Note that p_{worst_A} is actually the worst move for Prisoner A but the best move for Prisoner B. If Prisoner A knows Prisoner B's strategy, Prisoner A might be tempted to play "Cooperate" which could lead to the worst outcome for Prisoner A. But if Prisoner A plays "Defect", Prisoner A can ensure the worst outcome is avoided:

$$[p_{int} \mapsto D]^A \leadsto_A \neg p_{worst_A}.$$

This means Prisoner's A strategy can be reduced with the ability to ensure avoidance of the worst outcome:

$$Strat^A \equiv ([p_{int} \mapsto D]^A)$$

Without cooperation player A still have the strategy set $Strat^A \leadsto_B (p_{dom} \vee p_{worst_A})$. Note that p_{worst_A} is actually the worst move for Prisoner A but the best move for Prisoner B. If Prisoner A knows Prisoner B's strategy, Prisoner A might be tempted to play "Cooperate" which could lead to the worst outcome for Prisoner A. Prisoner A can ensure the worst outcome is avoided if Prisoner A plays "Defect".

In theory, in a repeated game the prisoners can enforce punishment and rewards since the players with interact multiple times. This means each can force the other player to play "cooperate". The problem comes with the sequence being finite; which means the last game played between the players results cooperation being unenforceable. In the last round the prisoner B has the following strategy:

$$[turn_B \wedge (< D^+ > leaf \vee < C^+ > leaf) \mapsto D]^B$$

This is worst case for prisoner A if they play cooperate in every round of the game: $[turn_A \mapsto C]^A \leadsto_A [turn_B \wedge (< D^+ > leaf \vee < C^+ > leaf) \mapsto D]^B$. But if prisoner A plays defect then they can enforce throughout the game the result will not

be the worst outcome: $[turn_A \mapsto C]^A \leadsto_A [turn_B \wedge (<D^+ > leaf \vee <C^+ > leaf) \mapsto D]^B \equiv \neg p_{worst_A}$.

Using backward induction on the game model, the result for the repeated version of the Prisoner's Dilemma is the same as for the single observation version. It results in Prisoner A choosing to enforce that the outcome is not the worst possible.

$$[turn_A \mapsto D]^A \leadsto_A \neg p_{worst_A}.$$

C.7 Summary

This Appendix discusses the application of the temporal hypergame framework to the classical game theoretic repeated Prisoner's Dilemma. Using the temporal hypergame framework, the concept of backwards induction was shown as a solution to the Prisoner's Dilemma. The Prisoner's Dilemma is symmetric and does not have differences in perception, therefore the structural properties do not allow for detailed hypergame analysis. The Prisoner's Dilemma is used for justification by example to show validity to the classical game theoretic problems. There is nothing in the formation of the temporal hypergame framework to limit its applicability to other classical game theoretic games.

Appendix D: Analysis of 'Hypergames and Bayesian Games'

Yasuo Sasaki and Kyoichi Kijima published "Hypergames and Bayesian Games: A Theoretical Comparison of the Models of Games with Incomplete Information". It appears in the Journal of Systems Science and Complexity [313]. In this article the authors make the claim "any hypergame can naturally be reformulated in terms of Bayesian games in an unified way". This claim is much stronger than the method they actually propose. There are limitations that results in hypergames that cannot be reformulated in terms of a Bayesian game. The authors discuss the limitations of their method, which limits the ability to reformulate a hypergame in terms of a Bayesian game. The purpose of this discussion is to cover what they did, including the claims they made, present information the authors missed and highlight the usefulness of hypergames as original proposed by P.G. Bennett [34] and extended later by Russell Vane [356].

D.1   Hypergames and Bayesian Games

Sasaki and Kijima propose a Bayesian Representation of Hypergames by using Harsanyi's theory that any game of incomplete information can be transformed into a game of complete information. Before discussing Sasaki and Kijima's work, an overview of Harsanyi research will be given for understanding. Harsanyi researched games of incomplete information in game theory. In game theory, a game of incomplete information is when partial or no information concerning the opponent's past moves are given in advance of the player's decision. Harsanyi claims uncertainties in the game and perceptual differences between players can be modeled as a game of complete information, where all players know the strategies and payoffs of every player, by:

- Players - Participation by an player is converted into the players action set. If the player is suppose to be out of the game, then the player is only allowed one action, "non-participation".

- Actions - Feasibility of a particular action for an player is converted into the players utility function. If the action is not feasible then the player receives a low utility whenever the action is taken.

- Utility Functions - Uncertainties in the game and perceptual differences between players are reduced to uncertainties or perceptual differences about the utility functions. Each possible utility function is then modeled as a type of player.

Hypergames are games of incomplete information by design. In a hypergame, one player may information or a strategy the other player(s) is not aware of or discounts. Because hypergames are games of incomplete information, Sasaki and Kijima applies Harsanyi's claim to hypergames as follows to transform the game from incomplete information to complete information.

- Set of Players Transformation - If player A does not believe another player B participates in the game, but player B actually does participate in the game, then **according to Harsanyi**, player A excluding player B from the game is the same as player A including player B in the set of players in the game but only allowing player B one action "non-participation". This allows every player to see a common set of players; a requirement for a game of complete information.

- Set of Actions Transformation - If player A does not believe an action is feasible for player B in the game, but the action is really feasible for player B (player A discounts an action of player B), then **according to Harsanyi**, player A excluding the action for player B is the same as player A including the action

for player B in the action set but assigning a very low utility to the action's usage for player B, in player A's own view of the game.

D.2   Issues with Modeling Hypergames as Bayesian Games

The authors clearly state there are a few issues with modeling hypergames as Bayesian games. This results in the inability to reformulate "any" hypergame in terms of a Bayesian game as the authors claim.

First, analyzing hypergames using Bayesian games is possible if one is only concerned with the hyper Nash equilibrium and Bayesian Nash equilibrium concepts. A Nash equilibrium is a set of actions in which neither player can increase their utility by unilaterally changing his or her strategy. If a player uses mixed strategies, then the expected value of the payoffs are maximized. A hyper Nash equilibrium [195], is an outcome where each player chooses an action that leads to a Nash equilibrium. The set of all hyper Nash equilibria for a given game, is the set of all possible outcomes "likely to happen" [195]. The authors do discuss that other equilibrium concepts may not allow for modeling as a Bayesian game [313].

Second, Bayesian games make an assumption that allows every player to see a common set of possibilities concerning the game structure. The authors discuss how in real situations, this assumption is "hard to accept", pointing out it is a controversial issue in epistemic game theory [313]. This leads the authors to conclude the assumption is incompatible with hypergames [313].

In real situations, it may not be possible for every player to see a common set of possibilities of the game structure. The inability to see a common set of possibilities may be due to the fact that in real world situations information is missing, obstructed, or misleading. For example, the U.S. Government uses all three techniques to limit and/or the release of sensitive information to adversaries. The U.S. Government will classify information to create missing information on true capabilities, redact

documents to obstruct information, and participate in deception techniques to mislead adversaries about true capabilities.

D.3   Hypergame Normal Form

Sasaki and Kijima only apply Harsanyi's claims to the original hypergame model developed by P.G. Bennett [34]; they do not discuss or mention the extension to hypergame theory by Russell Vane in his doctorial dissertation published in 2000. Vane extends the original hypergame model by including an assessment of the player's beliefs, as well as an assessment about the risk of selecting a strategy that is a non Nash equilibrium mixed strategy (NEMS). This extended hypergame model is called Hypergame Normal Form [356].

Vane uses six assessment properties when building Hypergame Normal Form. The first three are taken from game theory and decision theory without modification [356].

> Property 1: The result of every Row strategy versus every considered Column strategy must be estimable. Which means every row-column pair leads to an outcome.

> Property 2: Every result can be evaluated as a utility for Row.

> Property 3: Every result can be evaluated as Row's quantification of the Column's utility.

Vane then proposes three additional assessment properties as he constructs the Hypergame Normal Form in order to allow Row to record beliefs about the reasoning of Column. These three are unique to Vane's research [356].

> Property 4: Row has a reasoning context about the Column's decision process, which allows the recording of Row's information about Column's expected play.

Property 5: Row has a way to record uncommitted subgame belief about Column's expected play.

Property 6: Row has a way to express and quantify his exposure to Column's most effective counterstrategy. This property corresponds to the fear of being outguessed in the game.

Vane extended hypergame theory by building upon the theoretical concepts in game theory, decision theory concepts, and risk mitigation knowledge [356]. Vane research aims to correct some of the shortfalls of game theory and decision theory by creating a mathematical bridge between the two. Why is a mathematical bridge needed? Game theory and decision theory disagree about how the to access probability of a given situation arising and how to use the probability to select the most desirable strategy.

The probabilities are implicitly derived from an expert's view of the game being modeled in game theory, referred to as the full game. In game theory, the selected strategy is often the strategy with the least amount of vulnerability. This is seen through the use of Nash equilibriums as game theoretic solutions, where neither player can increase their utility by unilaterally changing his or her strategy. A Nash equilibrium leads to a game theoretic solution with vulnerability minimized by discouraging an opponent to change their choice of action (doing so would result in a decrease in expected utility by the Nash equilibrium definition).

Decision theory derives the probabilities by understanding the situation. These probabilities are normally derived without regard for the opponent (by definition decision theory only deals with the current player not the opponent), resulting in the player's best guess being used to select strategies. This leads to selected strategies with higher expected utility, when compared to game theory, but one a smart and

clever adversary would be able to exploit or outmaneuver (an example would be to use game theoretic concepts to exploit the decision theoretic outcome).

Hypergame Normal Form offers an approach that minimizes the risk of being outmaneuvered by an adversary while achieving a higher expected utility. This is accomplished by incorporating both explicit and implicit assessments of the probability distributions. This allows the risky strategy selection (although resulting in higher expected utility) from decision theory to be compared to the safer, but less rewarding strategies recommended in the game theoretic approach using the full game - where a player considers not only their actions on the outcome, but also the actions of their opponent(s).

Introduction of hypergame expected utility by Vane is the mathematical bridge between game theory and decision theory. Vane does this by taking into account the player's fear of being outguessed or outmaneuvered by their opponent. When there is no fear of being outguessed, then hypergame expected utility resembles the expected utility from decision theory. When the fear of being outguessed approaches 1 (100%), the hypergame expected utility is the worst case solution. For the worst case solution game theory provides a better solution than decision theory, since game theory leads to maximizing the minimum expected utility.

D.4 Advantages of Hypergames

Hypergame theory, even with its foundation in game theory, is fundamentally heretical to game theoretic concepts. Game theory assumes full knowledge of the conflict and a common mode of rationality among players in order to derive consistent alignment in beliefs. This consistent alignment in beliefs under lays the assumption of rationality in game theory. This leads to a player believing opponents reasons in the same manner as themselves. Hypergame theory disregards consistent alignment because it is often a fallacy given the wide range of beliefs within human nature. For

246

example, terrorists may not reason according to rational expectations. A rational terrorist would try to maximize the expected utility (in terms of causalities) of carrying out an attack, but from experience terrorist often carry out attacks that result in lower expected utility (in terms of causalities) but are still successful. This means a terrorist's beliefs being any success maximizes expected utility, while a rational actor, such as law enforcement, would expect maximization of expected utility in terms of causalities to be rational.

Hypergames allow a player to take advantage of the of strategies resulting the highest expected utility, while minimizing being deceived by an opponent. In extended hypergames this is accomplished by comparing the game theoretic solution against the decision theoretic solution. Any solution that is greater than, equal to, or less is considered completely effective, partially effective, or ineffective strategies. Vane discounts the expected utility by evaluating the vulnerability resulting from its use and assigning a weight to it.

D.5   Bayesian Games and Probability Distributions

As discussed previously, when a hypergame is reformulated as a Bayesian game a probability distribution is required. Hypergames, as extended by Vane, do not require an initial probability distribution as do the Bayesian games. Even the original hypergame model proposed by P.G. Bennett does not require an initial probability distribution. When a hypergame is transformed into a Bayesian game, a probability distribution is required, but none of the decision theory axioms determine how the probability distribution is derived.

In reality, this means additional information is need (not just the hypergame model) in order to represent a hypergame in decision theory. This additional information is used to derive the probability distribution. If additional information is required on the Bayesian model than was previously given in the hypergame model,

247

did the authors actually present a method to transform hypergames into a Bayesian game and ultimately decision theory? Or did the authors just propose a Bayesian representation of hypergames if more information is given.

It could be argued that game theoretic concepts such as the Nash equilibrium can be used to derive the probability distribution. While this is true this would not be a pure decision theory based model; it would still be a game theoretic and raises the question if the hypergames have been fully modeled under decision theory without the need for game theory.

D.6   Summary

This appendix discussed the paper by Yasuo Sasaki and Kyoichi Kijima published in 2012 on hypergames and Bayesian games. The authors work is discussed, their claims, as well as weaknesses of their approach including the limited usage of equilibrium concepts and inability to see a common game structure in real world situations. Questions are also raised as to if hypergames, which do not require a probability distribution, can be transformed into a Bayesian game without additional information.

Bibliography

[1] Abad-Santos, Alexander. "China Is Winning the Cyber War Because They Hacked U.S. Plans for Real War". *The Atlantic Wire*, May 2013.

[2] Abadi, M. "The power of temporal proofs". *Theoretical Computer Science*, 65(1):35 – 83, 1989.

[3] AFP. "Cyber attacks a grwoing threat for US financial system". Online, May 2015. URL http://phys.org/news/2015-05-cyber-threat-financial.html.

[4] Agotnes, Thomas, Wiebe van der Hoek, and Michael Wooldridge. "Temporal qualitative coalitional games". *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, AAMAS '06, 177–184. ACM, New York, NY, USA, 2006. ISBN 1-59593-303-4. URL http://doi.acm.org/10.1145/1160633.1160662.

[5] Aiello, Marco, Ian Pratt-Hartmann, and Johan Benthem. "What is Spatial Logic?" Marco Aiello, Ian Pratt-Hartmann, and Johan Benthem (editors), *Handbook fo Spatial Logics*, 1 – 11. Springer Netherlands, 2007.

[6] Alazzawe, Anis, Asad Nawaz, and Murad Mehmet Bayraktar. "Game Theory and Intrusion Detection Systems". *ISA 767 - Secure E-Commerce*, 2006.

[7] Alchourron, C., P. Gardenfors, and D. Makinson. "On the logic of theory change: partial meet contraction and revision functions". *The Journal of Symbolic Logic*, 50:510 – 530, 1993.

[8] Alexander, David. "U.S. defense firms face relentless cyberattacks". *Reuters*, September 2011.

[9] de Alfaro, L., M. Faella, T. Henzinger, R. Majumdar, and M. Stoelinga. "The element of surprise in timed games". *14th international Conference on Concurrency Theory*, volume 2761, 144 – 158. LNCS, Springer-Verlag, 2003.

[10] Aliseda, Atocha. *Abductive Reasoning*, 2 – 225. In Abductive Reasoning Logical Investigations into Discovery and Explanation. Springer, April 2006.

[11] Aliseda, Atocha. "Abductive Reasoning: Challenges Ahead". *THEORIA. An International Journal for Theory, History and Foundations of Science*, 22(3):261 – 270, 2009.

[12] Allen, J.F. "Maintaining knowledge about temporal intervals". *ACM Communications*, 26(11):832 – 843, November 1983.

[13] Allen, J.F. "Towards a general theory of action and time". *Artificial Intelligence*, 23:123 – 154, 1984.

[14] Allen, J.F. and P.J. Hayes. "A common sense theory of time". *9th International Joint Conference on Artificial intelligence (IJCAI)*, 528 – 531. August 1985.

[15] Alpcan, Tansu and Tamer Basar. "An Intrusion Detection Game with Limited Observations". *Proceedings of 12th International Symposium on Dynamic Games and Applications*. July 2006.

[16] Alur, R., T. Feder, and T.A. Henzinger. "The benefits of relaxing punctuality". *Proceedings of the 10th Annual ACM Symp. on Princ. Dist. Comp.* 1991.

[17] Alur, R. and T.A. Henzinger. "A really temporal logic". *Proceedings 30th IEEE Symp. on Foundations of Computer Science*, 164 – 169. 1989.

[18] Alur, R. and T.A. Henzinger. "Real-time logics: Complexity and expressiveness". *Proceedings 5th IEEE Symp. on Logic in Computer Science*, 492 – 501. 1990.

[19] Alur, R., T.A. Henzinger, and O. Kupferman. "Alternating time temporal logic". *Jornal of the ACM*, 49:672 – 713, 2002.

[20] An, B., J. Pita, E. Shieh, M. Tambe, C. Kiekintveld, and J. Marecki. "Guards and protect: Next generation applications of security games". *SIGECOM*, 31 – 34. 2011.

[21] Aumann, R. J. and M. Maschler. "Game theoretic analysis of a bankruptcy problem from the talmud". *Journal Economic Theory*, 36:195 – 213, 1985.

[22] Bananno, Giacomo. "Temporal Interaction of Information and Belief". *Studia Logica: An International Journal for Symbolic Logic*, 86(3):375 – 401, August 2007.

[23] Bandoyopadhyay, Tridib and Reda Sebalia. "Countering Cyber Terrorism: Investment Models Under Decision and Game Theoretic Frameworks". *Proceedings of the Southern Association for Information Systems Conference*. Richmond, VA, March 2008.

[24] Banieqbal, B. and H. Barringer. *A study of an extended temporal language and a temporal fixed point calculus*. Technical Report UMCS-86-10-2, Department of Computer Science, University of Manchester, November 1986.

[25] Baral, C. and M. Gelfond. "Logic programming and reasoning about actions". M. Fisher, D. Gabbay, and L. Vila (editors), *Handbook of Temporal Reasoning in Artificial Intelligence, Foundations of Artificial Intelligence*, volume 1, 389 – 428. Elsevier Press, 2005.

[26] Baron, Jonathan. *Thinking and Deciding.* Cambridge University Press, 4th edition, October 2007.

[27] Barringer, H. and D. Gabbay. "Modal varieties of temporal logic". M. Fisher, D. Gabby, and L. Vila (editors), *Handbook of Temporal Reasoning in Artificial Intelligence*, volume 1, 119 – 166. Elsevier, 2005.

[28] Barringer, H., R. Kuiper, and A. Pnueli. "A compositional temporal approach to a CSP-like language". *The Role of Abstract Models in Information Processing.* IFIP Working Conference, Vienna, 1985.

[29] Barringer, H., R. Kuiper, and A. Pnueli. "A really abstract concurrent model and its temporal logic". *13th ACM Symposium on the Principles of Programming Languages (POPL).* January 1986.

[30] Basilico, N., N. Gatti, and F. Amigoni. "Leader-follower strategies for robotic patrolling in environments with arbitrary topologies". *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems*, 500 – 503. 2009.

[31] Battigalli, P. "Strategic Independence and Perfect Bayesian Equilibria". *Journal of Economic Theory*, 70:201 – 234, 1996.

[32] Bennett, Michael and Russell R. Vane III. "Using Hypergames for Deception Planning and Counter deception Analysis". *Defense Intelligence Journal*, 15:117 – 138, 2006.

[33] Bennett, Peter and Steve Cropper. "Uncertainty and Conflict: Combining Conflict Analysis and Strategic Choice". *Journal of Behavioral Decision Making*, 3:29 – 45, 1990.

[34] Bennett, P.G. "Toward a theory of hypergames". *OMEGA*, 5(6):749 – 751, 1977.

[35] Bennett, P.G. "Bidders and Dispenser: manipulative hypergames in a multinational context". *European Journal of Operational Research*, 4:293 – 306, 1980.

[36] Bennett, P.G. "Hypergames, the development of an approach to modeling conflicts". *Futures*, 12:489 – 507, 1980.

[37] Bennett, P.G., S. Cropper, and C. Huxham. *Rational Analysis for a Problematic World*, chapter Modelling Interactiv Decisions: The Hypergame Focus. John Wiley and Sons, Chichester, UK, 1989.

[38] Bennett, P.G., S. Cropper, and C. Huxham. *Rational Analysis for a Problematic World*, chapter Using theHypergame Perspective: A Case Study, 315 – 340. John Wiley and Sons, Chichester, UK, 1989.

[39] Bennett, P.G. and M.R. Dando. "Complex strategic analysis: A hypergame perspective of the Fall of France". *Journal of the Operational Research Society*, 30(1):23 – 32, 1979.

[40] Bennett, P.G. and M.R. Dando. "Fall Gelb and other games: A hypergame perspective of the fall of france, 1940". *Journal of the Conflict Research Society*, 1(2):1 – 32, 1979.

[41] Bennett, P.G. and M.R. Dando. "The Arms Race as a Hypergame". *Futures*, 293–306, 1982.

[42] Bennett, P.G., M.R. Dando, and R.G. Sharp. "Using Hypergames to Model Difficult Social Issues: An Approach to the case of soccer hooliganism". *Journal of the Operational Research Society*, 31:621 – 635, 1980.

[43] Bennett, P.G. and C.S. Huxham. "Hypergames and what they do a 'soft O.R.' approach". *Journal of the Operational Research Society*, 33:41 – 50, 1982.

[44] Bennett, P.G., C.S. Huxham, and M.R. Dando. "Shipping in crisis - a trial run for 'Live' application of the hypergame approach". *Omega*, 9:579 – 594, 1981.

[45] van Benthem, J.F.A.K. *The Logic of Time.* Reidel, 1983.

[46] van Benthem, Johan. "Games in dynamic epistemic logic". *Bulletin of Economic Research*, 53(4):219 – 248, 2001.

[47] Berger, James O. *Statistical Decision Theory and Bayesian Analysis.* Springer Series in Statistics. Springer, 2nd edition, 1993.

[48] Bernoulli, D. "Exposition of a New theory of the Measurement of Risk". *Econometrica*, 22:23–36, 1954.

[49] Bertrand, J. "Théorie mathématique de la richesse sociale". *Journal des Savants*, 499 – 508, 1883.

[50] van Binsbergen, Jules H. and Leslie M. Marx. "Exploring Relations Between Decision Analysis and Game Theory". *Decision Analysis*, 4(1):32 – 40, March 2007.

[51] Blackburn, P., M. de Rijke, and Y. Venema. *Modal Logic.* Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2001.

[52] Blass, A. and Y. Gurevich. "Existential fixed-point logic". *Computation Theory and Logic, Lecture Notes in Computer Science*, 270:20 – 36, 1987.

[53] Bonanno, G. "Rational Belief Equilibria". *Economic Notes*, 22:430 – 463, 1993.

[54] Bonanno, G. "A simple modal logic for belief revision." *Synthese (and knowledge, Rationality and Action)*, 147:193 – 228, 2005.

[55] Bonanno, Giacomo. "Branching Time, Perfect Information Games, and Backward Induction". *Games and Economic Behavior*, 36(1):57 – 73, 2001. ISSN 0899-8256. URL http://www.sciencedirect.com/science/article/pii/S0899825699908126.

[56] Borel, E. "La theorie du jeu et les equations integrales a noyau symetrique". *Comptes Rendus de l'Academie des Sciences*, 173:1304–1308, 1921.

[57] Bowman, H. and S. Thompson. "A decision procedure and complete axiomatization of finite interval temporal logic with projection". *Journal of Logic and Computation*, 13(2):195 – 239, 2003.

[58] Brumley, Lachlan. *HYPANT: A Hypergame Analysis Tool.* Honors thesis, Monash University, November 2003.

[59] Bryant, J.W. "Hypergmaps: a Representation of Perceptions in Conflicts". *Omega*, 11:575 – 586, 1983.

[60] Burgess, J.P. "Axioms for tense logic; 1 - 'Since' and 'until'". *Notre Dame Journal of Formal Logic*, 23(4):367 – 374, October 1982.

[61] Burgess, J.P. and Y. Gurevich. "The decision problem for linear temporal logic". *Notre Dame Journal of Formal Logic*, 26(2):115 – 128, 1985.

[62] Burke, David A. *Towards a game theory model of information warfare.* Master's thesis, Air Force Institute of Technology, WPAFB, Ohio, 1999.

[63] Bursztein, E. "Extending anticipation games with location, penalty and timeline". *Proceedings of the 5th International Workshop on Formal Aspects in Security and Trust (FAST)*, 272 – 286. Lecture Notes in Computer Science, Springer-Verlag, October 2008.

[64] Bursztein, E. "Using strategy objectives for network security analysis". *Proceedings of the 4th International Conferences on Information Security and Cryptology (INSCRYPT)*. Lecture Notes in Computer Science, Springer, 2008.

[65] Bursztein, E. and J. Goubault-Larrecq. "A logical framework for evaluating network resilience against faults and attacks". *12th annual Asian Computing Science Conference (ASIAN)*, 212 – 227. Springer-Verlag, December 2007.

[66] University of California, Berkeley. "Cyber-Physical Systems âĂŞ a Concept Map", March 2016. URL http://CyberPhysicalSystems.org. Accessed 3 March 2016.

[67] Carroll, T. E. and D. Grosu. "A game theoretic investigation of deception in network security". *ICCCN'09*, Proceedings of the 18th International Conference on Computer Communications and Networks, 1 – 6. 2009.

[68] Chaib-draa, Brahim. "Hypergame analysis in multiagent environments". *In Proc. AAAI Spring Symp.*, 147 – 149. 2001.

[69] Chamberlin, Edward. "Duopoly-Value Where Sellers are Few". *Quarterly Journal of Economics*, 44(1):63 – 100, 1929.

[70] Chaochen, Z. and M.R. Hansen. *Duration Calculus - A Formal Approach to Real-Time Systems.* EATCS Monographs in Theoretical Computer Science. Springer, 2004.

[71] Chellas, B.F. *Modal Logic: An Introduction.* Cambridge University Press, 1980.

[72] Chen, Lin and Jean Leneutre. "A Game Theoretical Framework on Intrusion Detection in Heterogeneous Networks". *IEEE Transactions on Information Forensics and Security*, 4(2), June 2009.

[73] Chen, T.M. and S. Abu-Nimeh. "Lessons from Stuxnet". *Computer*, 44(4):91–93, 2011.

[74] Clark, Sandy, Matt Blaze, David Nelson-Fisher, and Matthew Elmore. "Inside the OODA Loop - Towards an Aggressive Defense". *ShmooCon 2012*, 27–29. 2012. Abstract.

[75] Cohn, A. "Spatial Reasoning". F. van Harmelen, V. Lifschitz, and B. Porter (editors), *Handbook of Knowledge Representation.* Elsevier, 2008.

[76] College, Butte. *Deductive, Inductive, and Abductive Reasoning.* Tip sheet, Butte College, ¡http://butte.edu/departments/cas/tipsheets/thinking/reasoning.html¿, Accessed 27 May 2013.

[77] contributors, Wikipedia. "List of games in game theory — Wikipedia, The Free Encyclopedia", 2016. URL https://en.wikipedia.org/w/index.php?title=List_of_games_in_game_theory&oldid=699637358. [Online; accessed 05-February-2016].

[78] Conway, J.H. *On Numbers and Games.* A K Peters Ltd, second edition, 2001. (first edition by Academic Press, 1976).

[79] Cournot, A. "Recherches sur les Principes Mathematiques de la Theorie des Richesses". *Researches into the Mathematical Principles of the Theory of Wealth*, 1987.

[80] Cryptography and System Security Lab. *sKyWIper (a.k.a Flame a.k.a Flamer): A complex malware for targeted attacks.* Technical Report 1.05, Budapest University of Technology and Economics, May 2012.

[81] Cybenko, George. "Cyber Adversary Dynamics". Online, February 2013. URL http://www.dtic.mil/dtic/tr/fulltext/u2/a579667.pdf.

[82] DARPA-JFACC. *Deception in non-cooperative games with partial information*, Proceedings of the 2nd DARPA-JFACC Symposium on Advances in Enterprise Control, 2000.

[83] Davis, E. "Infinite loops in finite time: Some Observations". *International Conference on Knowledge Representation and Reasoning (KR)*, 47 – 58. 1992.

[84] of Defense, The Department. *The Department of Defense Cyber Strategy.* The Department of Defense, April 2015.

[85] of Defense, U.S. Dept. *Joint Doctrine for Information Operations.* Joint Publication 3-13, United States Government, Oct 1998.

[86] Dictionaries, Oxford. "Cyber". Online, 2016. URL http://www.oxforddictionaries.com/us/definition/american_english/cyber.

[87] Ditmarsch, Hans P. *Knowledge Games.* Ph.D. thesis, University of Amsterdam, November 2000.

[88] Dixit, Avinash and Susan Skeath. *Games of Strategy.* W.W. Norton and Company, 2nd edition, 2004.

[89] Drakengren, T. and P. Jonsson. "Processing qualitative temporal constraints". M. Fisher, D. Gabbay, and L. Vila (editors), *Handbook of Temporal Reasoning in Artificial Intelligence, Foundations of Artificial Intelligence*, volume 1. Elsevier Press, 2005.

[90] Dresher, M. *The Mathematics of Games of Strategy: Theory and Applications.* Prentice-Hall, Englewood Cliffs, NJ, 1961.

[91] Dugatkin, Lee. *Game Theory and Animal Behavior.* Oxford University Press, 1998.

[92] Edgeworth, F. "La Teoria pura del monopolio". *Giornale degli Economisti*, 13 – 31, 1897.

[93] ElAlfi, Atta Ebrahim E., M. Esmel ElAlami, and Y. M. Asem. "Knowledge Extraction for Discriminating Male and Female in Logical Reasoning from Student Model". *CoRR*, abs/0911.0028, 2009.

[94] Emerson, E.A. "Temporal and modal logic". J. van Leeuwen (editor), *Handbook of Theoretical Computer Science*, 995 – 1072. Elsevier Science Publishers, 1989.

[95] Emerson, E.A. *Handbook of Theoretical Computer Science*, chapter Temporal Modal Logic, 997 – 1071. Elsevier Science Publishers, 1990.

[96] Emerson, E.A. and J.Y. Halpern. "'Sometimes' and 'not never' revisited: On branching time versus linear time". *Journal of the ACM*, 33:151 – 178, 1986.

[97] Faily, S. and I. Flechais. "User-Centered Information Security Policy Development in a Post-Stuxnet World". *Availability, Reliability and Security (ARES), 2011 Sixth International Conference on*, 716–721. 2011.

[98] Fidler, D.P. "Was Stuxnet an Act of War? Decoding a Cyberattack". *Security Privacy, IEEE*, 9(4):56–59, 2011.

[99] FireEye. *APT28: A Window into Russia's Cyber Espionage Operations.* Report, FireEye, 2014.

[100] Fishburn, P.C. *Utility Theory and Decision Making.* John Wiley and Sons, New York, 1970.

[101] Fisher, M. "A normal form for temporal logic and its application in theoremproving and execution". *Journal of Logic and Computation*, 7(4), July 1997.

[102] Fisher, M. "Foundations of Artificial Intelligence". M. Fisher, D. Gabbay, and L. Vila (editors), *Handbook of Temporal Reasoning in Artificial Intelligence*, volume 1. Elsevier, 2005.

[103] Fisher, Michael. "Temporal Representation and Reasoning". F. van Harmelen, V. Lifschitz, and B. Porter (editors), *Handbook of Knowledge Representation.* Elsevier, 2008.

[104] Flood, M.M. *Some Experimental games.* Research memorandum, RAND Corporation, Santa Monica, CA, 1952.

[105] Fraser, Niall M. and Keith W. Hipel. "Dynamic Analysis of the Cuban Missile Crisis". *Conflict Management and Peace Science*, 6(2), 1983.

[106] Fraser, Niall M. and Keith W. Hipel. *Conflict Analysis: Models and Resolutions.* Series Volume II: North-Holland Series in System Science and Engineering, Andres P. Sage, Editor. North-Holland, 1984.

[107] Fraser, Niall M., Keith W. Hipel, and Jose R. del Monte. "Approaches to Conflict Modeling: A Study of a Possible USA - USSR nuclear Confrontation". *Journal of Policy Modeling*, 5(3):397–417, 1983.

[108] Fraser, N.M. and K.W. Hipel. "Solving complex conflicts". *IEEE Transactions on Systems, Man and Cybernetics*, 9:805 – 815, 1979.

[109] Fraser, N.M. and K.W. Hipel. *Computer Analysis of a labor-management conflict.* Technical Report 71-SM270880, Department of Systems Design Engineering, University of Waterloo, 1980.

[110] Fraser, N.M. and K.W. Hipel. "Conflict analysis and bargaining". *Proc. 1980 International Conference on Cybernetics and Society, sponsored by the IEEE Systems, Man and Cybernetics Society*, 225 – 229, 1980.

[111] Fraser, N.M. and K.W. Hipel. "Computer assistance in labor-management negotiation". *Interfaces*, 11:22 – 30, 1981.

[112] Friedman, James W. "A Non-cooperative Equilibrium for Supergames". *The Review of Economic Studies*, 38(1):1–12, 1971. URL http://restud. oxfordjournals.org/content/38/1/1.short.

[113] Friend, J.K. and A. Hickling. *Planning under Pressure*. Pergamon, Oxford, 1987.

[114] Fudenberg, D. and J. Tirole. "Perfect Bayesian Equilibrium and Sequential Equilibrium". *Journal of Economic Theory*, 53:236 – 260, 1991.

[115] Fudenberg, Drew and Jean Tirole. *Game Theory*. The MIT Press, first edition, August 1991.

[116] Funk and Wagnells Standard Dictionary. *Conflict*. Fitxhenry and Whiteside, Toronto, 1974.

[117] Gabbay, D. *Investigations in Modal and Tense Logics with Applications to Problems in Philosophy and Linguistics*. Reidel, 1976.

[118] Gabbay, D. "Expressive functional completeness in tense logic". U. Monnich (editor), *Aspects of Philosophical Logic*, 91 – 117. Reidel, Dordrecht, 1981.

[119] Gabbay, D. and I. Hodkinson. "An axiomitization of the temporal logic with until and since over the real numbers". *Journal of Logic and Computation*, 1(2):229 – 259, 1990.

[120] Gabbay, D., A. Pnueli, S. Shelah, and J. Stavi. "The temporal analysis of fairness". *7th ACM Symposium on the Principles of Programming Languages (POPL)*, 163 – 173, January 1980.

[121] Gabbay, D., A. Pnueli, S. Shelah, and Y. Stavi. "Completeness results for the future fragment of temporal logic", 1980. Manuscript.

[122] Gabbay, D., M. Reynolds, and M. Finger. *Temporal Logic: Mathematical Foundations and Computational Aspects*, volume 2. Clarendon Press, Oxford, 2000.

[123] Gabby, D., I. hodkinson, and M. Reynolds. *Temporal Logic: Mathematical Foundations and Computational Aspects*, volume 1. Clarendon Press, Oxford, 1994.

[124] Gabelaia, D., R. Knotchakov, A. Kurucz, F. Wolter, and M. Zakharyaschev. "Combining spatial and temporal logics: expressiveness vs. complexity". *Journal of Artificial Intelligence Research (JAIR)*, 23:167 – 243, 2005.

[125] Galton, A.P. "A critical examination of Allen's theory of action and time". *Artificial Intelligence*, 42:159 – 188, 1990.

[126] Galton, A.P. "An investigation of 'Non-intermingling' principles in temporal logic". *Journal of Logic and Computation*, 6(2):271 – 294, 1996.

[127] Gardenfors, P. *Knowledge in flux: modeling the dynamics of epistemic states.* MIT Press, 1988.

[128] Gharesifard, Bahman and Jorge Cortés. "Evolution of the perception about the opponent in hypergames". *Proc. IEEE Conf. Decision Control*, 1076 – 1081, 2010.

[129] Gharesifard, Bahman and Jorge Cortés. "Learning of equilibria and misperceptions in hyper-games with perfect observations". *American Control Conference*, 4045 – 4050. 2010.

[130] Gharesifard, Bahman and Jorge Cortés. "Exploration of misperceptions in hypergames". *2011 49th Annual Allerton Conference*, 1565 – 1570. Communication, Control, and Computing (Allerton), September 2011.

[131] Gharesifard, Bahman and Jorge Cortés. "Stealthy strategies for deception in hypergames with asymmetric information". *Decision and Control and European Control Confer-ence (CDC-ECC)*, 5762 – 5767. IEEE, December 2011.

[132] Gharesifard, Bahman and Jorge Cortés. "Evolution of Players' Misperceptions in Hypergames Under Perfect Observations". *IEEE Transactions on Automatic Control*, 57(7):1627 – 1640, July 2012.

[133] Gharesifard, Bahman and Jorge Cortés. "Stealthy deception in hypergames under information-al asymmetry", 2013. Unpublished Draft.

[134] Ghosh, Sujata, Ben Meijering, and Rineke Verbrugge. "Logic meets cognition: empirical reasoning in games". *Proceedings of the 3rd International Workshop on Logics for Resource Bounded Agents (LRBA 2010)*, 627:15–34, 2010.

[135] Gibson, Alan S. *Applied Hypergame Theory for Network Defense.* Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, June 2013.

[136] Giesen, M.O. *The Hypergame Model - a Critical Examination Using Case-Studies.* Dissertation, University of Sussex, Department of Operational Research, 1978.

[137] Giesen, M.O. and P.G. Bennett. "Aristotle's Fallacy: A hypergame in the oil shipping business..." *Omega*, 7(4):309 – 320, 1979.

[138] Gilboa, I. "Questions in decision theory". *Annual Reviews in Economics*, 2:1–19, 2010.

[139] Goldblatt, R. *Logics of time and computation*, volume 7 of *CSLI Lecture Notes*. Stanford University, 1987.

[140] Goranko, V. "Coalition Games and Alternating Temporal Logics". *Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge*, TARK '01, 259–272. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001. ISBN 1-55860-791-9. URL http://dl.acm.org/citation.cfm?id= 1028128.1028157.

[141] Goranko, V., A. Montanari, and G. Sciavicco. "A road map of interval temporal logics and duration calculi". *Journal of Applied Non-Classical Logics*, 14(1 − 2):9 − 54, 2004.

[142] Gorman, Siobhan, August Cole, and Yochi Dreazen. "Computer Spies Breach Fighter-Jet Project". *The Wall Street Journal*, April 2009.

[143] Goto, Yuichi, Takahiro Koh, and Jingde Cheng. "A General Forward Reasoning Algorithm for Various Logic Systems with Different Formalizations". *12th International Conference, Knowledge-Based Intelligent Information and Engineering Systems, Proceedings Part II*, 526 − 535, September 2008.

[144] Graham, Ian, Fiona O'Doherty, Alan McKinnon, and Lynne Baxter. "Hypergame analysis of the stability of relationships between computer-based logistics systems". *International Journal of Production Economics*, 26(2):303 − 310, 1992.

[145] Grant, Simon and Timothy Van Zandt. "Expected Utility Theory". Paul Anand, Prasanta Pattanaik, and Clemens Puppe (editors), *Handbook of Rational and Social Choice*. Oxford University Press, 2008.

[146] Greenberg, Andy. "Hack Brief: Health Insurer Excellus says Attackers Breached 10M Records". Online, September 2015. URL http://www.wired.com/2015/09/ hack-brief-health-insurance-firm-excellus-says-attackers-breached-10m-records/.

[147] Greevini, A. "Processing qualitative temporal constraints". M. Fisher, D. Gabbay, and L. Vila (editors), *Handbook of Temporal Reasoning in Artificial Intelligence, Foundations of Artificial Intelligence*, volume 1. Elsevier Press, 2005.

[148] Griffin, C. and K. Moore. "A Framework for Modeling Decision Making and Deception with Semantic Information". *IEEE Symposium on Security and Privacy Workshops*, 68 − 74. IEEE Computer Society, 2012.

[149] Guelev, D. "A complete proof system for first-order interval temporal logic with projection". *Journal of Logic and Computation*, 14(2):215 – 249, 2004.

[150] Guelev, D. and D. van Hung. "A relatively complete axiomatisation of projection onto state in the duration Calculus". *Journal of Applied Non-Classical Logics*, 14(1 – 2):149 – 180, 2004.

[151] Gunes, Mesut. "Verification and Validation of Simulation Models", March 2016. URL http://www.mi.fu-berlin.de/inf/groups/ag-tech/teaching/2012_SS/L_19540_Modeling_and_Performance_Analysis_with_Simulation/10.pdf. Accessed 3 May 2016.

[152] Haack, S. *Philosophy of Logics*. Cambridge University Press, 1978.

[153] Halpern, J., Z. Manna, and B. Moszkowski. "A hardware semantics based on temporal intervals". *International Colloquium on Automata Languages and Programming (ICALP), Lecture Notes in Computer Science*, 278 – 291. Springer-Verlag, 1983.

[154] Halpern, J. and Y. Shoham. "A propositional modal logic of time intervals". *ACM Journal*, 38(4):935 – 962, 1991.

[155] Halpern, J.Y. and Y. Shoham. "A Propositional Modal Logic of Time Intervals". *IEEE LICS*, 279 – 292, 1986.

[156] Hamandawana, H., Raban Chanda, and Frank Eckardt. "Hypergame Analysis and Hydroconflicts in the Okavango Drainage Basin". *International Water Resources Association*, 32(4):538 – 557, 2007.

[157] Hansson, Sven Ove. *Decision Theory A Brief Introduction*. Technical report, Royal Institute of Technology, Stockholm, August 2005.

[158] Harel, E., O. Lichtenstein, and A. Pnueli. "Explicit clock temporal logic". *Proceedings 5th IEEE Symp. on Logic in Computer Science*, 402 – 413. 1990.

[159] Harrenstein, Paul, Wiebe van der Hoek, John-Jules Meyer, and Cees Witteven. "A modal characterization of Nash equilibrium". *Fundamenta Informaticae*, 57(2-4):281 – 321, 2003.

[160] Harsanyi, J.C. "Games with incomplete information played by Bayesian players". *Management Science*, 159–182, 320–334, 486–502, 1967.

[161] Hastie, Reid and Robyn M. Dawes. *Rational Choice in an Uncertain World - The Psychology of Judgment and Decision Making*. SAGE Publications, 2nd edition, November 2009.

[162] He, Fei, Jun Zhuang, and Nageswara S. V. Rao. "Game-Theoretic Analysis of Attack and Defense in Cyber-Physical Network Infrastructures". G. Lim and J.W. Hermann (editors), *Proceedings of 2012 Industrial and Systems Engineering Research Conference*. 2012.

[163] Henzinger, T., Z. Manna, and A. Pnueli. "Temporal proof methodologies for real-time systems". *Proceedings 18th ACM Symp. on Principles of Programming Languages*, 353 – 366. 1991.

[164] Henzinger, T. and V. Prabhu. "Timed alternating-time temporal logic". *Formats 2006*, 4202:1 – 18, 2006.

[165] Hewett, Rattikorn, Sudeeptha Rudrapattana, and Phongphun Kijsanayothin. "Cyber-Security Analysis of Smart Grid SCADA Systems with Game Models". *9th Cyber and Information Security Research Conference*, 109 – 112. ACM, April 2014.

[166] Hipel, K.W., A. Dagnino, and N.M. Fraser. "A Hypergame Algorithm for Modeling Misperceptions in Bargaining". *Journal of Environmental Management*, 27(2):131 – 152, 1988.

[167] Hipel, K.W., M. Wang, and N.M Fraser. "Hypergame analysis of the Falkland Island crisis". *International Studies Quarterly*, 32:335 – 358, 1988.

[168] Hirsch, R. "From points to intervals". *Journal of Applied Non-Classical Logics*, 4(1):7 – 27, 1994.

[169] Hirsch, R. "Relation algebras of intervals". *Artificial Intelligence*, 83:1 – 29, 1996.

[170] Hodkinson, I. and M. Reynolds. "Separation-past, present, and future". *We Will Show Them! Essays in Honour of Dov Gabbay*, 2:117 – 142, 2005.

[171] Hodkinson, I., F. Wolter, and M. Zakharyaschev. "Decidable fragments of first-order temporal logics". *Annals of Pure and Applied Logic*, 106:85 – 134, 2000.

[172] van der Hoek, Wiebe, Wojciech Jamroga, and Michael Wooldridge. "A Logic for Strategic Reasoning". *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '05, 157–164. ACM, New York, NY, USA, 2005. ISBN 1-59593-093-0. URL http://doi.acm.org/10.1145/1082473.1082497.

[173] Honsell, Furio and Marina Lenisa. "Conway Games, Algebracially and Coalgebraically". *Logical Methods in Computer Science*, 3(8):1 – 30, 2011.

[174] House, James Thomas and George Cybenko. "Hypergame Theory Applied to Cyber Attack and Defense". Edward M. Carapezza (editor), *Sensors, and*

Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IX, volume 7666. 2010.

[175] Howard, N. *Paradoxes of Rationality, Theory of Metagames and Political Behavior.* M.I.T. Press, Cambridge, MA, 1971.

[176] Hughes, G.E. and M.J. Cresswell. *An Introduction to Modal Logic.* Methuen, New York, 1968.

[177] Huxham, C. and P. Bennett. "Floating Ideas - An Experiment in Enhancing Hypergames with Maps". *Omega*, 13(4):331 – 347, 1985.

[178] Huxham, C.S. and P.G. Bennett. "Hypergames and Design Decisions". *Design Studies*, 4(4):227 – 232, October 1983.

[179] III, Russell R. Vane. "Hypergame Theory for DTGT Agents". *American Association for Artificial Intelligence*, 2000.

[180] III, Russell R. Vane. "Planning for Terrorist-Caused Emergencies". M.E. Kuhl, N.M. Steiger, and J.A. Joines (editors), *Proceedings of the 2005 Winter Simulation Conference.* 2005.

[181] III, Russell R. Vane. "Advances in Hypergame Theory". *International Journal of Autonomous Agents and Multi-Agent Systems*, 2006.

[182] Inohara, T., S. Takashashi, and B. Nakano. "Integration of games and hypergames generated from a class of games". *Journal of the Operational Research Society*, 48:423 – 432, 1997.

[183] IV, Russell Vane. *Veridian Security Policy Assistant Users Manual.* Users manual, U.S. Department of Defense, July 2002.

[184] Jain, M., J. Tsai, J. Pita, C. Kiekintveld, S. Rathi, M. Tambe, and F. Ordonez. "Software assistants for randomized patrol planning for the lax airport police and the federal air marshal service". *Interfaces*, 40:267 – 290, 2010.

[185] Jain, S., R.R. Creasey, J. Himmelspach, K.P. White, and M. Fu (editors). *Verification and Validation of Simulation Models*, Proceedings of the 2011 Winter Simulation Conference. IEEE, 2011.

[186] Jetley, Raoul and Ben Chelf. "Diagnosing Medical Device Software Defects Using Static Analysis. Medical Device & Diagnostic Industry", 2009. URL http://www.devicelink.com/mddi/archive/09/05/006.html. Accessed 24 May 2016.

[187] Johnson, M., F. Fang, R. Yang, M. Tambe, , and H. Albers. "Patrolling to maximize pristine forest area". *Proc. of the AAAI Spring Symposium on Game Theory for Security, Sustainability, and Health.* AAAI, 2012.

[188] Jones, B.D. "Bounded rationality". *Annual Review of Political Science*, 2:297 – 321, 1999.

[189] Kamp, J. *Tense Logic and the Theory of Linear Order.* Ph.D. thesis, University of California, Los Angeles, May 1968.

[190] Kanazawa, T. and T. Ushio. "Multi-Population Replicator Dynamics with Changes of Interpretations of Strategies". *International Symposium on Nonlinear Theory and its Appli-cations.* October 2005.

[191] Kanazawa, T., T. Ushio, and T. Yamasaki. "Replicator dynamics of evolutionary hypergames". *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, 3828–3833. 2003.

[192] Kanazawa, Takafuni, Toshimitsu Ushio, and Tatsushi Yamasaki. "Replicator Dynamics of Evolutionary Hypergames". *IEEE Transactions on Systems, Man, and Cybernetics âĂŞ Part A: Systems and Humans*, 37(1):132 – 138, 2007.

[193] Kartik, Navin. "Lecture Notes for 1st Year Ph.D. Game Theory", March 2009. URL http://econweb.ucsd.edu/~jsobel/200Cs09/Kartik_Notes.pdf. Accessed 29 January 2016.

[194] Kesten, Y., Z. Manna, and A. Pnueli. "Temporal verification of simulation and refinement". *A Decade of Concurrency, Reflections and Perspectives, REX School/Symposium, Lecture Notes in Computer Science*, 803, 1994.

[195] Kijima, K. "An intelligent ploy-agent learning model and its application". *Information and Systems Engineering*, 2:47 – 61, 1996.

[196] Kivimaa, Juri, Andres Ojamaa, and Enn Tyugu. "Graded security expert system". R. Setola and S. Geretshuber (editors), *CRITIS 2008, LNCS 5508*, 279 – 286. Springer-Verlag, Berlin Heidelberg, 2009.

[197] Klimas, Liz. "Did China Get Plans For Out Semi-Secret F-22 Raptor and Copy It?" *The Blaze*, June 2012.

[198] Kopp, Carlo. "Shannon, Hypergames and Information Warfare". W. Hutchinson (editor), *Proceedings of the 3rd Australian Information Warfare and Security Conference (IWAR).* 2002.

[199] Kopp, Carlo. "Hypergames vs Information Conflict Strategies", 2006. URL http://www.csse.monash.edu.au/~carlo/InfoWar/Lectures/CSE-468-06.pdf. Accessed 24 May 2016.

[200] Korzhyk, D., V. Conitzer, and R. Parr. "Complexity of computing optimal stackelberg strategies in security resource allocation games". *Proc. of the 24th AAAI Conference on Artificial Intelligence*, 805 – 810. 2010.

[201] Kowalski, R. and M. Sergot. "A logic-based calculus of events". *New Generation Computing*, 1(4):67 – 95, 1986.

[202] Koymans, R. and W.-P. de Roever. "Examples of a real-time temporal logic specification". M.I. Jackson and M.J. Wray (editors), *The Analysis of Concurrent Systems*, Lecture Notes in Computer Science 207, 231 – 252. Springer-Verlag, 1983.

[203] Koymans, R., J. Vytopyl, and W.-P. de Roever. "Real-time programming and asynchronous message passing". *Proceedings 2nd ACM symp. on Princ. of Dist. Comp.*, 187 – 197, 1983.

[204] Kreps, David M. and Robert Wilson. "Sequential Equilibria". *Econometrica*, 50:863 – 894, 1982.

[205] Kripke, S. "A semantical analysis of modal logic I: normal propositional calculi". *Zeitschrigt fur Mathematische Logik und Grundlagen der Mathematik*, 9:67 – 96, 1963.

[206] Kripke, S.A. "Semantical analysis of modal logic I: normal propositional calculi". *Z. Math. Logik Grund. Math*, 9:67 – 96, 1963.

[207] Kuelh, D. "From cyberspace to cyber power". F.D. Kramer, S.H. Starr, and L. Wentz (editors), *Cyberpower and national Security.* Potomac Books, 2009.

[208] Lab, Kaspersky. *Operation "Red October": Indicators of Compromise and Mitigation Data.* Whitepaper 1.4, Kaspersky Lab, January 2013.

[209] Laboratory, Los Alamos National. *Concepts of Model Verification and Validation.* Technical Report LA-14167-MS, Department of Energy, October 2004.

[210] Labs, Kaspersky. ""Red October" Diplomatic Cyber Attacks Investigation", January 2023. Accessed June 2013.

[211] Ladkin, P. "The completeness of a natural system for reasoning with time intervals". *10th International Joint Conference on Artificial Intelligence (IJCAI)*, 462 – 467. August 1987.

[212] Ladkin, P. and R. Maddux. "On binary constraint problems". *ACM Journal*, 41:435 – 469, 1994.

[213] Lamport, L. "Sometimes is Sometimes "Not Never" on the temporal logic of programs". *7th Annual ACM Symp. on Principles of Programming Languages*, 171 – 185, 1980.

[214] Lane, Robert. "Introduction to Philosophy: Lecture Notes: University of West Georgia", June 2013. Unpublished.

[215] Langner, Ralph. "Stuxnet: Dissecting a Cyberwarfare Weapon". *Security Privacy, IEEE*, 9(3):49–51, 2011.

[216] Laroussinie, F., N. Markey, and Ph. Schnoebelen. "Temporal logic with forgettable past". *17th IEEE Symposium on Logic in Computer Science (LICS)*, 383 – 392. IEEE Computer Society, 2002.

[217] Leclerc, Maxime and Brahim Chaib-draa. "Hypergame Analysis in E-Commerce: A Preliminary Report". *Scientific Series, CIRANO*. 2002.

[218] Lemke, C.E. and J.T. Howson. "Equilibrium Points of Bimatrix Games". *SIAM Journal on Applied Mathematics*, 12(2):413 – 423, 1964.

[219] Lennon, Mike. "Hackers Hit 100 Banks in 'Unprecedented' $1 Billion Cyber Heist: Kaspersky Lab". Online, February 2015. URL http://phys.org/news/2015-05-cyber-threat-financial.html.

[220] Levine, David K. *What is Game Theory*. http://levine.sscnet.ucla.edu/general/whatis.htm, UCLA, October 2013.

[221] Lewis, James A. "Cyberwar Thresholds and Effects". *IEEE Security and Privacy*, October 2011.

[222] Lichtenstien, O., A. Pnueli, and L. Zuck. "The glory of the past". *Lecture Notes in Computer Science*, 196 – 218. Workshop on Logics of Programs, Springer, 1985.

[223] Ligozat, G. "Weak representation of interval algebras". *8th American national Conference on Artificial intelligence (AAAI)*, 715 – 720. 1990.

[224] Ligozat, G. "Tractable relations in temporal reasoning: pre-convex relations". *ECAL Workshop on Spatial and Temporal Reasoning*. August 1994.

[225] Lin, F. "Situation Calculus". F. van Harmelen, V. Lifschitz, and B. Porter (editors), *Handbook of Knowledge Representation*. Elsevier, 2008.

[226] Luyendijk, Joris. "Cyber attacks could be bigger threat to our banking system than bad debts". Online, Oct 2015. URL http://www.theguardian.com/sustainable-business/2015/oct/24/cyber-attacks-could-be-bigger-threat-to-our-banking-system-than-bad-debts.

[227] Ma, J. and B. Knight. "Reified temporal logics: an overview". *Artificial Intelligence Review*, 15(3):189 – 217, 2001.

[228] Macaulay, S. "Non-contractual Relation in Business: A Preliminary Study". *American Sociological Review*, 28:55 – 67, 1963.

[229] Manna, Zohar and Amir Pnueli. *The temporal logic of reactive and concurrent systems.* Springer-Verlag, 1992.

[230] Marchi, Ezio. "A note on temporal games". *Zeitschrift fÃijr Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 12(2):98–106, 1969. ISSN 0044-3719. URL http://dx.doi.org/10.1007/BF00531643.

[231] Marchi, Ezio. "A class of games that evolve". *Zeitschrift fÃijr Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 18(4):271–280, 1971. ISSN 0044-3719. URL http://dx.doi.org/10.1007/BF00535029.

[232] Marsh, S. and M.R. Dibben. "Trust, untrust, distrust and mistrust - an exploration of the dark (er) side". *Trust Management*, 3477:17 – 33, 2005.

[233] Marx, M. and M. Reynolds. "Undecidability of compass logic". *Journal of Logic and Computation*, 9:897 – 914, 1999.

[234] Mas-Colell, Andreu, Michael D. Whinston, and Jerry R. Green. *Microeconomic Theory.* Oxford University Press, New York, 1995.

[235] Mateski, Mark E., Thomas A. Mazzuchi, and Shahram Sarkani. "The Hypergame Perception Model: A Diagrammatic Approach to Modeling Perception, Misperception, and Deception". *Military Operations Reserach*, 15(2):21 – 37, 2010.

[236] Mathews, Anna Wilde. "Anthem: Hacked Database Included 78.8 Million People". Online, Febuary 2015. URL http://www.wsj.com/articles/anthem-hacked-database-included-78-8-million-people-1424807364.

[237] McCaney, Kevin. "NSA Looks to protect defense contractors from cyberattack". *GCN*, June 2011.

[238] McDermott, D. "A temporal logic for reasoning about processes and plans". *Cognitive Science*, 6:101 – 155, 1982.

[239] McKelvey, Richard D., Andrew M. McLennan, and Theodore L. Turocy. "Gambit: Software Tools for Game Theory", 2010. URL: http://www.gambit-project.org.

[240] McTaggart, J.M.E. *The Nature of Existence*, volume 1. Cambridge, 1927.

[241] Mertens, J.-F. and A. Neyman. "Stochastic games". *International Journal of Game Theory*, 10(2):53–66, 1981. ISSN 0020-7276. URL http://dx.doi.org/10.1007/BF01769259.

[242] Mertens, J.F., S. Sorin, and S. Zamir. "Repeated Games". *CORE DP*, 1–519, 1989.

[243] Monderer, Dov and Moshe Tennenholtz. "Distributed games". *Journal of Economic Literature*, Games and Economic Behavior:55 – 72, 1999.

[244] Mookherjee, Dilip and Barry Sopher. "Learning Behavior in an Experimental Matching Pennies Game". *Games and Economic Behavior*, 7:62 – 91, 1994.

[245] Moszkowski, B. *Reasoning about digital circuits.* Ph.D. thesis, Computer Science Department, Stanford University, 1983.

[246] Moszkowski, B. "Compositional reasoning about projected and infinite time". *1st IEEE International Conference on Engineering of Complex Computer Systems (ICECCS)*, 238 – 245. IEEE Computer Society Press, 1995.

[247] Moszkowski, B.C. *Reasoning about Digital Circuits.* Ph.D. thesis, Stanford University, 1983.

[248] Myerson, Roger B. *Game Theory Analysis of Conflict.* Harvard University Press, Cambridge, Massachusetts, 1997.

[249] Nakashima, Ellen. "Confidential report lists U.S. weapons system designs compromised by Chinese cyberspies". *The Washington Post*, May 2013.

[250] Nash, J. "Non-Cooperative Games". *Annals of Mathematics*, 54(2):286–295, 1951.

[251] Nash, J.F. "The bargaining problem". *Econometrica*, 18(2):155 – 162, 1950.

[252] Nash, J.F. "Equilibrium points in N person games". *Proceedings of the National Academy of Sciences*, 48 – 49. 1950.

[253] Nash, J.F. "Two-person cooperative games". *Exonometrica*, 21(1):128 – 140, January 1953.

[254] Nau, Dana. "Introduction to Game Theory: Repeated Games", March 2016. URL fromhttps://www.cs.umd.edu/users/nau/game-theory/7%20Repeated%20games.pdf. Accessed 29 May 2016.

[255] Nayak, A., M. Pagnucco, and P. Peppas. "Dynamic belief revision operators". *Artificial Integlligence*, 146, 2003.

[256] von Neumann, J. "Zur theorie der gesellschaftsspiele". *Mathematische Annalen*, 100:295–320, 1928.

[257] von Neumann, J. "Zur theorie der gesellschaftsspiele". R.D. Luce and A.W. Tucker (editors), *Contributions to the Theory of Games IV*, 13–42. Princeton University Press, 1959.

[258] von Neumann, J. and O. Morgenstern. *Theory of Games and Economic Behavior.* Princeton University Press, second edition, 1944.

[259] von Neumann, John and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 2nd edition, 1947.

[260] Novani, Santi and Kyoichi Kijima. "Symbiotic Hypergame Analysis of Value Co-Creation Process in Service System". *IEEE*, 2010.

[261] Obama, Barack. "Critical Infrastructure Security and Resilience". *The White House - Presidental Policy Directive*, February 2013. URL http://www.whitehouse.gov/the-press-office/2013/02/12/presidential-policy-directive-critical-infrastructure-security-and-resil.

[262] Obama, Barack. "Improving Critical Infrastructure Cybersecurity". *The White House - Executive Order*, February 2013. URL http://www.whitehouse.gov/the-press-office/2013/02/12/executive-order-improving-critical-infrastructure-cybersecurity.

[263] Obama, Barack. "Blocking the Property of Certain Persons Engaging in Significant Malicious Cyber-Enabled Activities". *The White House - Executive Order*, April 2015. URL https://www.whitehouse.gov/the-press-office/2015/04/01/executive-order-blocking-property-certain-persons-engaging-significant-m.

[264] Obama, Barack. "Promoting Private Sector Cybersecurity Information Sharing". *The White House - Executive Order*, February 2015. URL https://www.whitehouse.gov/the-press-office/2015/02/13/executive-order-promoting-private-sector-cybersecurity-information-shari.

[265] Ochs, Jack. "An Experimental Study of Games with Unique Mixed Strategy Equilibria". *Games and Economic Behaviour*, 10:202 – 217, 1995.

[266] Okada, Norio, K.W. Hipel, and Y. Oka. "Hypergame Analysis of the Lake Biwa Conflict". *Water Resources Research*, 21(7):917 – 926, 1985.

[267] for Operations Research, The Washington Institute and the Management Sciences. "Feedback". *Newsletter*, 27(2):1 – 6, March 2006. Www.informs.org/content/download/266804/2524773/file/feedback-vol27no2.pdf.

[268] Ordonez, F., M. Tambe, J.F. Jara, M. Jain, C. Kiekintveld, and J. Tsai. *Handbook on Operations Research for Homeland Security*, chapter Deployed security games for patrol planning. International Series in Operations Research and Management Science. Springer, 2012.

[269] Osborne, M.J. *An introduction to game theory*. Technical report, Oxford University Press, August 2003.

[270] Osborne, M.J. and A. Rubinstein. *A Course in Game theory*. MIT Press, Cambridge, Massachusetts, 1994.

[271] Ostroff, J.S. *Temporal Logic for Real-Time Systems.* Advanced Software Development Series. John Wiley and Sons, 1989.

[272] Ozdaglar, Asu. "6.254 : Game Theory with Engineering Applications Lecture 16: Repeated Games âĂŞ II", April 2010. URL http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/ 6-254-game-theory-with-engineering-applications-spring-2010/lecture-notes/ MIT6_254S10_lec16.pdf. Accessed 5 February 2016.

[273] Pagliery, Jose. "UCLA Health hacked, 4.5 million victims". Online, July 2015. URL http://money.cnn.com/2015/07/17/technology/ucla-health-hack/.

[274] Parikh, Rohit. "The logic of games and its applications". *Annals of Discrete Mathematics*, 24:111 – 140, 1985.

[275] Paul, Soumya, R. Ramanujam, and Sunil Simon. "Stability under Strategy Switching". *LNCS 5635*, Proceedings 5th Conference on Computability in Europe (CiE 2009), 389 – 398. Springer, 2009.

[276] Pauly, Marc. *Logic for Social Software.* Ph.D. thesis, University of Amsterdam, October 2001.

[277] Peirce, C.S. *Pragmatism as a Principle and Method of Right Thinking*, volume 5 of *Harvard Lectures on Pragmatism.* State University of New York Press, 1903. Paragraphs 188 – 189.

[278] Peirce, C.S. "A Neglected Argument for the Reality of God". *Hibbert Journal*, 7:90 – 112, 1908.

[279] Peirce, C.S. "On the Logic of drawing History from Ancient Documents especially from Testimonies". The Peirce Edition Project (editor), *The Essential Peirce: Selected Philosophical Writings, Volume 2*, 75 – 114. Indiana University Press, 1998.

[280] Peterson, Martin. *An Introduction to Decision Theory.* Cambridge Introductions to Philosophy. Cambridge University Press, May 2009.

[281] Pita, J., M. Tambe, C. Kiekintveld, S. Cullen, and E. Steigerwald. "Guards - game theoretic security allocation on a national scale". *Proc. of the 10th International Conference on Autonomous Agents and Multiagent Systems.* 2011.

[282] Pnueli, A. "Linear and Branching Structures in the Semantics and Logics of Reactive Systems". *Proceedings of the 12th ICALP*, 15 – 32. 1985.

[283] Pnueli, A. "Applications of temporal logic to the specification and verification of reactive systems: A survey of current trends". J.W. de Bakker, W.-P. de Roever, and G. Rozenberg (editors), *Current Trends in Concurrency*, Lecture Notes in Computer Science 224, 510 – 584. Springer=Verlag, 1986.

[284] Poundstone, W. *Prisoner's Dilemma.* Doubleday, NY, NY, 1992.

[285] on Cybersecurity for the 44th Presidency, CSIS Commission. *Securing cyberspace for the 44th presidency.* Technical report, Center for Strategic and International Studies, 2008.

[286] Price, J. Maynard Smithand G.R. "The logic of animal conflict". *Nature*, 246(5427):15 – 18, 1973.

[287] Prior, A. *Past, Present, and Future.* Clarendon Press, Oxford, 1967.

[288] Putro, U.S., K. Kijima, and S. Takahashi. "Adaptive learning of hypergame situations using a genetic algorithm". *IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans*, 13(5):562 – 572, 1985.

[289] Putro, U.S., K. Kijima, and S. Takahashi. "Simulation approach to learning problem in hypergame situation by genetic algorithm". *IEEE International Conference on Systems, Man, and Cybernetics*, 4:260 – 265, 1999.

[290] Putro, U.S., K. Kijima, and S. Takahashi. "Simulation of Adaptation Process in Hypergame Situation by Genetic Algorithm". *Systems Analysis Modeling Simulation*, 40(1), 2000.

[291] Qu, Youg, Yexin Song, and Jianiun Zhang. "An Outcome Preference Information Aggregation Model and Its Algorithm in Hypergame Situations". *Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, 494 – 497. 2008.

[292] Ramanujam, R. and S. Simon. "A Logical Structure for Strategies". Giacomo Bonanno, Wiebe van der Hoek, and Michael Wooldridge (editors), *Logic and the Foundations of Game and Decision Theory (LOFT 7)*, Texts in Logic and Games 3. Amsterdam University Press, Amsterdam, 2008.

[293] Rapoport, A. and A.M. Chammah. "The Game of Chicken". *American Behavioral Scientist*, 10, 1966.

[294] Ratliff, Jim. "Infinitely Repeated Game with Discounting", May 2016. URL http://www.virtualperfection.com/gametheory/5.2.InfinitelyRepeatedGames.1.0.pdf. Accessed 29 May 2016.

[295] Regents, UC. "Cyber-Physical Systems". Online, 2016. URL http://cyberphysicalsystems.org/.

[296] Reichgelt, H. and L. Vila. "Temporal qualification in artificial intelligence". M. Fisher, D. Gabbay, and L. Vila (editors), *Handbook of Temporal Reasoning in Artificial Intelligence, Foundations of Artificial Intelligence*, volume 1. Elsevier Press, 2005.

[297] Reid, Samuel. "On Generalizing a Temporal Formalism for Game Theory to the Asymptotic Combinatorics of S5 Modal Frames". *CoRR*, abs/1305.0064, 2013.

[298] Reiter, R. *Knowledge in Action.* MIT Press, 2001.

[299] Reitman, W.R. *Cognition and Thought.* Wiley, New York, 1965.

[300] Rescher, N. and A. Urquhart. *Temporal Logic.* Springer-Verlag, New Yorl, 1971.

[301] Resnik, Michael. *Choices - An Introduction to Decision Theory.* Univeristy of Minnesota Press, June 1987.

[302] Review, Cyberspace Policy. *Assuring a Trusted and Resilient Information and Communications Infrastructure.* Technical report, United States Government, 2009.

[303] Reynolds, M. "More past glories". *IEEE Symposium on Logic in Computer Science (LICS).* IEEE Press, 2000.

[304] Richelson, J.T. "Soviet Strategic Doctrine and Limited Nuclear Operations". *Journal of Conflict Resolution*, 23(2):326 – 336, 1979.

[305] Rosner, R. and A. Pnueli. "A choppy logic". *IEEE Symposium on Logic in Computer Science (LICS)*, 306 – 313. IEEE Computer Society, 1986.

[306] Roy, Sankardas, Charles Ellis, Sajjan Shiva, Dipankar Dasgupta, Vivek Shandilya, and Qishi Wu. "A Survey of Game Theory as Applied to Network Security". *Proceedings of the Hawaii International Conference on System Sciences.* 2010.

[307] Rudrapattana, Sudeeptha. *Cyber-Security Analysis in Smart Grid SCADA Systems: A Game Theoretic Approach.* Masters thesis, Texas Tech University, 2013.

[308] Said, A.K. and D.A. Hartley. "A Hypergame Approach to Crisis Decision-Making: The 1973 Middle East War". *The Journal of the Operational Research Society*, 33(10):937–948, 1982.

[309] Sanger, David E. and Nicole Perlroth. "Bank Hackers Steal Millions via Malware". Online, February 2015. URL http://www.nytimes.com/2015/02/15/world/bank-hackers-steal-millions-via-malware.html?_r=0.

[310] Sasaki, Y. *Modeling subjectivity and interpretations in games: A hypergame theoretic approach.* Ph.D. thesis, Tokyo Institute of Technology, March 2013.

[311] Sasaki, Y., N. Kobayashi, and K. Kijima. "Mixed Extension of Hypergames and Its Application to Inspection Games". *ISSS Proceedings of the 51st Annual Conference*, 777:1 – 9, 2007.

271

[312] Sasaki, Yasuo and Kyoichi Kijima. "Preservation of Misperceptions - Stability Analysis of Hypergames". *Proceedings of the 52nd Annual Meeting of the ISSS*, 1 – 15, 2008.

[313] Sasaki, Yasuo and Kyoichi Kijima. "Hypergames and Bayesian Games: A Theoretical Comparison of the Models of Games with Incomplete Information". *Journal of System Science Complex*, 25:720–735, 2012.

[314] Savage, L.J. *The Foundations of Statistics*. Dover Publications, New York, 2nd edition, 1954.

[315] Scarvalone, Mira. "Game Theory and the Minimax Theorem". URL http://www.math.uchicago.edu/~may/VIGRE/VIGRE2008/REUPapers/Scarvalone.pdf. Accessed 5 February 2016.

[316] Schwalbe, Ulrich and Paul Walker. "Zermelo and the Early History of Game Theory". *Games and Economic Behavior*, 34(1):123 – 137, 2001. ISSN 0899-8256. URL http://www.sciencedirect.com/science/article/pii/S0899825600907942.

[317] Schwartz, R.L., P.M. Melliar-Smith, and F.H. Vogt. "An interval-based temporal logic". E.M. Clarke and D.C. Kozen (editors), *Proceedings Workshop on Logics of Programs*, Lecture Notes in Computer Science 164, 501 – 512. Springer=Verlag, 1983.

[318] Scientific, Boston. "Validation, Verification, Qualification: Which is right and does it really matter?", 2006. URL http://www.mnasq.org/files/summaries/validation_verification_qualification.pdf. Accessed 24 May 2016.

[319] Scott, John. "Rational Choice Theory". G. Browning, A. Halcli, and F. Webster (editors), *Understanding Contemporary Society: Theories of The Present*. Sage Publications, 2000.

[320] Selten, R. "Spieltheoretische Behandlung eines Oligopolmodells mit Nachfrageträgheit". *Zeitschrift für die gesamte Staatswissenschaft*, 12:301 – 324, 1965.

[321] Selten, R. "What is bounded rationality?" *Proceedings of the Dahlen Conference*, 1 – 25. SFB Discussion Paper B-454, May 1999.

[322] Selten, Reinhard. "The chain store paradox". *Theory and Decision*, 9(2):127–159, 1978. ISSN 0040-5833. URL http://dx.doi.org/10.1007/BF00131770.

[323] Service, IDG News. *Iran confirms stuxnet hit centrifuges*. Report, IDG News Service, November 2010.

[324] Shapley, L. S. "Stochastic Games". *Proceedings of the National Academy of Sciences*, 39(10):1095–1100, 1953. URL http://www.pnas.org/content/39/10/1095.short.

[325] Shen, Dan, Genshe Chen, Jose B. Cruz Jr., Leonard Haynes, Martin Kruger, and Erik Blasch. *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications*, volume 6571, chapter A Markov Game Theoretic Data Fusion Approach for Cyber Situational Awareness. SPIE, 2007.

[326] Shiva, Sajjan, Sankardas Roy, and Dipankar Dasgupta. "Game Theory for Cyber Security". *Proceedings of the Cyber Security and Information Intelligence Research Workshop*. April 2010.

[327] Shoham, Yoav and Kevin Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, Cambridge, UK, 2009.

[328] Shupe, M.C., W.M. Wright, K.W. Hipel, and N.M. Fraser. "The nationalization of the Suez Canal - a hypergame analysis". *Journal of Conflict Resolution*, 24:477 – 493, 1980.

[329] Simon, H.A. *Models of Man*. Wiley and Sons, New York, 1957.

[330] Simon, H.A. "Theories of bounded rationality". C.B. McGuire and R. Radner (editors), *Deciion and Organization*, 161 – 176. North-Holland, 1972.

[331] Singh, Anshuman, Arun Lakhotia, and Andrew Walenstein. "Malware Anti-malware Games". *Proceedings of the International Conference on Information Warfare*, 319. January, 2010.

[332] Sistla, A.P., M.Y. Vardi, and P. Wolper. "The complementation problem for Büchi automata with applications to temporal logic". *Thoeretical Computer Science*, 49:217 – 237, 1987.

[333] Smith, Chris. "Hackers exploit Flash in one of the largest malware attacks in recent history". Online, 2015. URL http://www.foxnews.com/tech/2015/08/04/hackers-exploit-flash-in-one-largest-malware-attacks-in-recent-history.html.

[334] Smith, J. Maynard. *Evolution and the Theory of Games*. Cambridge University Press, 1982.

[335] Smith, J. Maynard and G.A. Parker. "The logic of asymmetric contests". *Animal Behaviour*, 24:159 – 175, 1976.

[336] Sober, E. *Philosophy of Biology*. Oxford University Press, 1992.

[337] Song, Y.X., M.Q Dai, and Y. Cui. "Military Conflict Decision Modeling Based on Fuzzy Hyper-game". *Proceedings of the 6th International Conference on Manage-ment*, 410 – 415, 2007.

[338] Song, Y.X., Z.J. Li, and Y.Q. Chen. "Fuzzy Information Fusion for Hypergame Outcome Preference Perception". *Intelligent Control and Automation*, 344:882 – 887, 2006.

[339] Song, Y.X., Y. Qu, and Z.R. Liu. "Fusion and automatic ranking of fuzzy outcome preferences information in hypergame models". *Proceedings of 2005 International Conference on Machine Learning and Cybernetics*, volume 5, 2711 – 2715. 2005.

[340] Stanovich, Keith. *Rationality and the Reflective Mind*. Oxford University Press, December 2010.

[341] Stilman, Boris, Vladimir Yakhnis, and Oleg Umanskiy. "Introduction to LG hypergames for practical wargaming". *Proc. SPIE 4716, Enabling Technologies for Simulation Science VI*, volume 386. 2002.

[342] Stilman, Boris, Vladimir Yakhnis, Oleg Umanskiy, and Maris McCrabb. "LG hypergames for effects-based operations". *Proc. SPIE 5091, Enabling Technologies for Simulation Science VII*, volume 426. 2003.

[343] Stokes, N.W. and K.W. Hipel. "Conflict analysis of an export credit trade dispute". *Omega*, 11(4):365 – 376, 1983.

[344] Surowik, Dariusz. "Temporal Logic Approach to Extensive Games". *Studies in Logic, Grammar and Rhetoric*, 7(20), 2004.

[345] Symantec. *W32.Duqu: The Precursor to the next Stuxnet*. Whitepaper 1.4, Security Response, November 2011.

[346] Szalas, A. "Arithmetical axiomatisation of first-order temporal logic". *Information Processing Letters*, 26:111 – 116, 1987.

[347] Szalas, A. and L. Holenderski. "Incompleteness of first-order temporal logic with until". *Theoretical Computer Science*, 57:317 – 325, 1988.

[348] Tait, Steven W. *The effects of budgetary constraints, multiple strategy selection, and rationality on equilibrium attainment in an information warfare simulation*. Master's thesis, Air Force Institute of Technology, WPAFB, Ohio, 2001.

[349] Thérien, D. and T. Wilke. "Nesting until and since in linear temporal logic". *Theory of Computing Systems*, 37(1):111 – 131, 2004.

[350] Tzu, Sun. *The Art of War*. English Translation and Commentary by Lionel Giles. El Paso Norte Press, special edition, 2005.

[351] Unknown. "Game Theory vs. Ordinary Decision Theory: I know that You Know that I Know..." *Scientific American*, May 2007. Http://www.scientificamerican.com/article.cfm?id=game-theory-vs-ordinary-decision.

[352] Vane, R. and P. Lehner. "Using hypergames to increase planned payoff and reduce risk". *Autonomous Agents Multi-Agent System*, 5:365 – 380, 2002.

[353] Vane, R.R. and P.E. Lehner. "Applications of Hypergame Theory to Adversarial Planning". *JDL Symposium Naval Postgraduate School*. Monterey, CA, 1990.

[354] Vane, R.R. and P.E. Lehner. "Hypergames and AI in Planning". *DARPA Research Symposium*. San Diego, CA, 1990.

[355] Vane, R.R. and P.E. Lehner. "Using hypergames to select plans in adversarial environments". S. Parsons and M.J. Wooldridge (editors), *Proceedings of the First Workshop on Game Theoretic and Decision Theoretic Agents*, 103 – 111. 1999.

[356] Vane, Russell R. *Using Hypergames to Select Plans in Competitive Environments*. Ph.D. thesis, George Mason University, 2000.

[357] Vardi, M.Y. and P. Wolper. "Reasoning about infinite computations". *Information and Computation*, 115(1):1 – 37, 1994.

[358] Vejandla, Pavan, Dipankar Dasgupta, Aishwarya Kaushal, and Fernando Nino. "Evolving gaming strategies for attacker-defender in a simulated network environment". *Proceedings of the 2010 IEEE Second Interna- tional Conference on Social Computing*, 889 – 896. SOCIALCOM '10, IEEE Computer Society, Washington, DC, USA, 2010.

[359] Venema, Y. "A logic with the chop operator". *Journal of Logic and Computation*, 1:453 – 476, 1991.

[360] Vincent, Thomas. *Evolutionary Game Theory, Natural Selection, and Darwinian Dynamics*. Cambridge University Press, 2005.

[361] Wang, Muhong, Keith W. Hipel, and Niall M. Fraser. "Modeling Misperceptions in Games". *Behavioral Science*, 33:207 – 223, 1988.

[362] Wang, Muhong, Keith W. Hipel, and Niall M. Fraser. "Solution Concepts in Hypergames". *Applied Mathematics and Computation*, 34:147 – 171, 1989.

[363] Weber, Robert, Boris Stilman, and Vladimir Yakhnis. "Extension of the LG hypergame to inner games played over the topology of competing mind nets". *Proc. SPIE 5805, En-abling Technologies for Simulation Science IX*, volume 224. 2005.

[364] Weber, Robert James. "Noncooperative Games". *Game Theory and its Applications*, volume 24, 83–125. Proceedings of Symposia in Applied Mathematics, 1981.

[365] Weisstein, Eric W. "Minimax Theorem". Online, 2016. URL http://mathworld. wolfram.com/MinimaxTheorem.html.

[366] Wikipedia, The Free Encyclopedia. "Repeated game", May 2016. URL https: //en.wikipedia.org/w/index.php?title=Repeated_game&oldid=711083813. Accessed 29 May 2016.

[367] Wolper, P. "Temporal logic can be more expressive". *Information and Control*, 56(1-2):72 – 99, 1983.

[368] Woods, John. "Ignorance and Semantic Tableaux: Aliseda on Abduction". *THEORIA. An International Journal for Theory, History and Foundations of Science*, 22(3), 2009.

[369] Wright, W.M., M.C. Shupe, N.M. Fraser, and K.W. Hipel. "A conflict analysis of the Suez Canal invasion of 1956". *Conflict Management and Peace Science*, 5:27 – 40, 1980.

[370] Yavin, Y. "Pursuit-evasion differential games with deception or interrupted observation". *Computers and Mathematics with Applications*, 13(1-3):191 – 203, 1987.

[371] Zeng, Xianhai, Yexin Song, and Bing Luan. "Interactive integration model of hypergames with fuzzy preference perceptions in multi-conflict situations". *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, volume 2, 646–650. Nov 2009.

[372] Zermelo, E. "Uber eine anwendung der mengenlehre auf die theorie des schachspiels". *Proceedings Fifth International Congress of Mathematicians*, volume 2, 501–504. 1913.

[373] Zhou, C., C.A.R. Hoare, and A.P. Ravan. "A calculus of durations". *Information Processing Letters*, 40(5):269 – 276, 1991.

[374] Zhou, Wu, Zhaofeng Chen, Jimmy Su, Jing Xie, and Heqing Huang. "SLEMBUNK: AN EVOLVING ANDROID TROJAN FAMILY TARGETING USERS OF WORLDWIDE BANKING APPS". Online, December 2015. URL https: //www.fireeye.com/blog/threat-research/2015/12/slembunk_an_evolvin.html.

[375] Zhou, Wu, Heqing Huang, Zhaofeng Chen, Jing Xie, and Jimmy Su. "SLEMBUNK PART II: PROLONGED ATTACK CHAIN AND BETTER-ORGANIZED CAMPAIGN". Online, January 2016. URL https://www.fireeye. com/blog/threat-research/2016/01/slembunk-part-two.html.

[376] Zhuang, J., V. M. Bier, and O. Alagoz. "Modeling secrecy and deception in a multiple-period attacker-defender signaling game". *European Journal of Operational Research*, 203(2):409 – 418, 2010.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
16–06–2016	Doctoral Dissertation	Sept 2010–June 2016

4. TITLE AND SUBTITLE

A Temporal Framework For Hypergame Analysis Of Cyber Physical Systems In Contested Environments

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Kovach, Nicholas S., CIV

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB, OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-DS-16-J-006

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Intentionally Left Blank

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution Statement A: Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES

The U.S. Government retains access rights to this work and has the right to disseminate this work for Government purposes.

14. ABSTRACT

Game theory is used to model conflicts between one or more players over resources. It offers players a way to reason, allowing rationale for selecting strategies that avoid the worst outcome. Game theory lacks the ability to incorporate advantages one player may have over another player. A meta-game, known as a hypergame, occurs when one player does not know or fully understand all the strategies of a game. Hypergame theory builds upon the utility of game theory by allowing a player to outmaneuver an opponent, thus obtaining a more preferred outcome with higher utility. Recent work in hypergame theory has focused on normal form static games that lack the ability to encode several realistic strategies. One example of this is when a player's available actions in the future is dependent on his selection in the past. This work presents a temporal framework for hypergame models. This framework is the first application of temporal logic to hypergames and provides a more flexible modeling for domain experts. With this new framework for hypergames, the concepts of trust, distrust, mistrust, and deception are formalized. While past literature references deception in hypergame research, this work is the first to formalize the definition for hypergames. As a demonstration of the new temporal framework for hypergames, it is applied to classical game theoretical examples, as well as a complex supervisory control and data acquisition (SCADA) network temporal hypergame. The SCADA network is an example includes actions that have a temporal dependency, where a choice in the first round affects what decisions can be made in the later round of the game. The demonstration results show that the framework is a realistic and flexible modeling method for a variety of applications.

15. SUBJECT TERMS

Cyber warfare, game theory, hypergame theory, decision theory

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Gary Lamont (ENG)
U	U	U	UU	298	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636 x4718 gary.lamont@afit.edu

Standard Form 298 (Rev. 8–98)
Prescribed by ANSI Std. Z39.18